Qiang Ma
Graduate School of Informatics,
Kyoto University

Query-Free Information Retrieval Based on Spatio-temporal Criteria and Content Complementation

Doctoral Dissertation Series of Tanaka Laboratory
Department of Social Informatics, Graduate School of Informatics,
Kyoto University

ABSTRACT

Rapid progress in web database technologies and digital broadcasting technologies has made it possible to provide a vast volume of information. Usually, users use keywords or keyword-based queries to acquire interesting information. Unfortunately, it's not easy to form such queries sometimes. For instance, it is not easy to specify the keywords used to find interesting ones from the incoming information such as new pages and news articles. Moreover, the conventional information retrieval and filtering systems return the relevant information to a user based on the exact matching or best matching (similarity, etc.) between the information and user's interest (query). Sometimes, this fashion could not satisfy our diverse information needs. For example, sometimes we want fresh information and sometimes we want some information describing our daily life or some information providing details on our interesting topics.

As one solution of these problems, in this thesis, we propose a novel query-free information retrieval mechanism for discovering information from diverse perspectives without need to form queries by using keywords. One of the notable features of this mechanism is that it rises higher than similarity information retrieval. That is to say, we can discover fresh, popular and urgent information by estimating its temporal (time-based) value. We also can discover information describing our regional and daily life by estimating its spatial (region-based) value. Moreover, we can discover complementary information of a given example for information augmentation. For instance, we can search supplementary information of a TV-program from the Web to provide details on the TV-program or describe it from different perspectives.

In this thesis, we propose the following concepts and methods on query-free information retrieval mechanism.

1. Query-free information retrieval based on temporal criteria

   In order to find valuable information from new web pages or news articles, we propose a new information retrieval method which considers the worth of a web page or an article compared with past web pages or articles, that is, their temporal criteria (*freshness, popularity, and urgency*). Based on these temporal criteria, we propose a change monitoring and notification system *WebSCAN* (Web Sites Change Analyzer and Notifier), which monitors and analyzes the changes of web sites and notifies *important changes* to users by a push-type delivery mechanism. We also propose a concept and a way to construct a *virtual TV channel*, which is a user-defined (virtual) channel from existing push-based Internet channels. Our virtual TV channel is (1) to filter content of multiple push-based Internet

channels, (2) to merge selected articles from different channels, and (3) to present them by a TV-program-like GUI.

2. Query-free information retrieval based on spatial criterion

   There are a lot of web pages, whose contents are 'local' and maybe only interest residents of a narrow region. In this thesis, we propose a spatial criterion (*localness*) to discover local information which describes about our daily and regional life. We compute the localness of a web page by 1) estimating its region dependency, and 2) estimating the ubiquitousness of topics described by it. We also propose an application system which filters searched web pages based on localness to acquire or exclude local information from the search results.

3. Query-free complementary information retrieval for information augmentation

   It becomes possible to acquire interesting information from diverse information sources to improve the quality and detailedness of information. In this thesis, we propose a novel complementary information retrieval method for information augmentation. The retrieved information is not just similar to the given example (web page we are browsing, TV-program we are watching, etc.), but also provides some additional information to detail it or describe it from different perspectives. In addition, we propose an application system which integrates TV-program and its complementary web pages in real time to augment the content of TV-program.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background and Motivation

The vast amount of information is available on the WWW. About 2.1 billion pages are available on the Web, and about 7 million new pages appear per day (2000/7)[Cyveillance, 2000]. At the same time, with the spreading of broadband services (Fiber To The Home (FTTH), Digital Subscriber Line (DSL), cable Internet, and wireless) and digital broadcasting technology, much attention is focused on data broadcasting systems via the Internet or digital broadcasting because of their potential and convenience. The concept of these systems is generally based on the push-type information delivery technology[Sumiya and Miyabe, 1999]. The push-type information delivery system does not require that users behave in an active manner in order to access information resources because the required and/or updated information is transmitted continuously and automatically. It has made it possible to provide a vast volume of information.

Everyone, not only the professionals but also the amateurs, can access these information sources and acquire favorite information from them. The number of Internet users in the world was estimated at more than 680 million at the end of November 10, 2003[Inc., 2003]. In Japan, there were 69.42 million Internet users at the end of 2002[Ministry of Public Management, Home Affairs Posts and Telecommunications, Japan, 2003] and the user population for broadband services was estimated to be 19.55 million as of the end of 2002.

The tremendous progress and spread of the information technologies have involved great changes in our life. For example, from the Internet and the broadcasting, we can find information about our work, shopping, health etc. Conventionally, users use keywords or keyword-based queries to acquire interesting information. Unfortunately, gathering interesting information is a difficult task for a novice user even if he uses the search engines or information filtering systems. For instance, it is not easy to specify the keywords used to find interesting information from the incoming information such as new pages and news article. The user must have experience and skill to form the query and find the relevant pages from the large number of documents returned. Moreover, the conventional information retrieval and filtering systems return the relevant infor-

mation to a user based on the matching or best matching (similarity, etc.) between the information and user's interest (the query). Sometimes, this fashion could not satisfy our diverse information needs, such as that we want fresh information sometimes, and sometimes we want information describing about our daily life or providing details on our interesting topic.

Digital television combines broadcasting and computer technologies into a powerful new medium and changes the way consumers watch TV[Digital Video Broadcasting Project, 2003]. On the other hand, with the spread of broadband services that provide high speed connection to the Internet, rich content (video, music, etc.) is available to view in real time. That is to say, the infrastructure for integration of TV-programs and the Internet is prepared. The intuitive integration of the TV and the Internet, such as synchronizing a TV program with its related web pages, should improve the quality of information and broaden our horizons because each complements the other. However, it could not interest use to synchronize a TV-program with its similar web pages, because the similar web pages could not give us additional information. We would like to browse web pages which can provide supplementary information on the TV-program. This is beyond the conventional information retrieval and filtering technologies.

As one solution of these problems, in this thesis, we propose a novel query-free information retrieval mechanism to discover information from diverse perspectives and without necessary to form queries by using keywords.

1.2 Overview

This research is focused on query-free information retrieval to select valuable information from the perspectives of time, space, and content augmentation. To estimate the temporal value of information, we define some temporal criteria such as *freshness*, *popularity* and *urgency*. We propose a criterion (*localness*) to estimate the spatial value of information. In other words, we use the spatial criterion *localness* to find the information describing about our daily and regional life. Moreover, from the content augmentation perspective, we propose a notion of *complementarity* to discover supplementary information about our interesting topics.

In this thesis, we intent to discuss mainly three research topics; namely, information retrieval based on temporal criteria, information retrieval based on spatial criterion, and complementary information retrieval for information augmentation.

1. Query-free information retrieval based on temporal criteria

   Broadcasting-type information dissemination systems on the Internet are becoming increasingly popular due to advances in the area of web technology and information delivery. One of the notable features of push-based, multiple-channel-based information dissemination systems is to send information to users in a form of time-series articles. Conventional information retrieval and filtering methods do not consider well the worth of an article from the standpoint of the time-series feature. We propose a new query-free information retrieval mechanism which considers the worth of an article compared with past delivered articles, that is, their time-series features (*freshness, popularity, and urgency*). We called these time-series features temporal criteria.

Based on these temporal criteria, we propose a concept and a way to construct a *virtual TV channel*, which is a user-defined (virtual) channel from existing push-based Internet channels. Our virtual TV channel is (1) to filter contents of multiple push-based Internet channels, (2) to merge selected articles from different channels, and (3) to present them by a TV-program-like GUI.

Moreover, based on these temporal criteria, we propose a change monitoring and notification system *WebSCAN* (Web Sites Change Analyzer and Notifier), which monitors and analyzes the changes of pre-registered web sites and notifies **important changes** to users by a push-type delivery mechanism.

2. Query-free information retrieval based on spatial criterion

   With the spreading of the Internet, information about our daily and regional life is becoming to be more and more active on the WWW (World Wide Web). That's to say, there are a lot of web pages, whose contents are 'local' and could only interest residents of a narrow region. The conventional information retrieval systems and search engines, such as Google[Google, 2003], Yahoo[Yahoo!, 2003], etc., are very useful to help users finding interesting information. However, it's not yet easy to find or exclude 'local' information about our daily and region life with the conventional information retrieval and filtering technologies. In this thesis, we propose a new spatial criterion *localness* to discover local information from the WWW. We compute the localness degree of a web page by 1) estimating its region dependence: the frequency of geographical words and the content coverage of this web page, and 2) estimating the ubiquitousness of its topic to estimate whether it is usual information that appears often in our daily life. Based on the notion of localness, we also propose a useful application which ranks the search results and returns local or not-local information to users.

3. Query-free complementary information retrieval for information augmentation

   With the spreading of digital broadcasting and broadband internet connection services, the infrastructure for integration of the TV and the Internet is prepared and we can find the additional information of a TV-program from the Web. That is to say, it becomes possible to acquire our interesting information from diverse information source and media to improve the quality and detailedness of information. In this thesis, we propose a novel complementary information retrieval method for information augmentation. One of the notable features of this method is that it can be used to find complementary information of a given web page or video. That is to say, the retrieved information is not just similar to the given web page or video. It can provide some additional information to detail the given one or describe it from different perspectives. We also show some evaluation results of the complementary information retrieval mechanism. In addition, we proposed an application system *WebTelop*, which integrates TV-program and its related web pages in real time to augment the content of the TV-program.

The main features of the query-free information retrieval mechanism can be summarized as follows:

- It goes beyond similar-information retrieval.
  The goal of our query-free information retrieval mechanism is not similar information, but fresh (or popular, or urgent), local and complementary information. Here, the complementary information means that can provide details on a given example or describing it from different perspectives.

- Forming query by using keywords is not necessary.
  The freshness, popularity and urgency are temporal criteria and localness is a spatial criterion. We could use them solely without any additional keyword. The complementary information is searched based on the content analysis automatically. Thus, users need not to form a query by using keywords. Of course, we also can use keywords in combination with these new criteria and methods.

- It retrieves information based on semantic criteria.
  The proposed criteria are meaningful and have been defined by comparisons between the new (or given) web page or article and the others from perspectives of time, space and content complementation.

This thesis is composed of five chapters including this introduction. In Chapter 2, we describe the query-free information retrieval method based on temporal criteria and its applications. In Chapter 3, we discuss the spatial criterion and the query-free information retrieval method based on it. In Chapter 4, we propose a complementary information retrieval method and its application system. We go over the research topics and state the conclusions of this thesis in Chapter 5.

CHAPTER 2

INFORMATION RETRIEVAL BASED ON TEMPORAL CRITERIA

2.1 Introduction

Recently, much attention is focused on data broadcasting systems via the Internet or digital broadcasting because of their potential and convenience. The concept of these systems is generally based on the push-type information delivery technology. The push-type information delivery does not require that users behave in an active manner in order to access information resources because the required and/or updated information is transmitted continuously and automatically. Users select their favorite channels provided by a service server in advance, and then the system will transmit information for each user at regular intervals. Recently, several push technologies, such as PointCast[PointCast Network, 1999] and Castanet[Marimba, 1998], have been developed. These systems are not only applied to transmit text data but also hypertext and hypermedia data. One of the notable features of push-based, multiple-channel-based information dissemination systems is to send information to users in a form of time-series articles. Conventional information retrieval and filtering methods do not consider well the worth of an article from the standpoint of the time-series feature.

On the other hand, the vast amount of information is available on the WWW. Usually, users use the bookmarks or automatic navigator software to access their favorite web sites to acquire valuable information. However, the Web is dynamic[Brewingtion and Cybenko, 2000], in other words, web pages are changed and web sites are created or disappear at any time and in arbitrary manner. Thus, it's not easy to acquire the fresh and valuable information timely.

To find interesting information for users from the large quantity of data, information filtering techniques and search engines, which are mainly based on the keywords, have been very useful. However, since the keywords of incoming news articles and new web pages are sometimes unknown, these typical methods may fail in acquiring the *fresh* (or *popular*) information. To acquire the fresh, popular and urgency information from the Web and the data broadcasting systems, in

this thesis, we propose some temporal criteria (time-series features), such as *freshness, popularity* and *urgency*.

Based on the time-series features (freshness, popularity and urgency), we propose two application systems: virtual TV channel and WebSCAN.

Virtual TV channel is a personal on-line news broadcasting system with new filtering/synthesizing and presentation approaches. The virtual TV channel is one broker to discover user interesting information, and to synthesize them as one virtual channel. The information is retrieved based on the similarity with a given user profile, freshness, popularity and urgency.

We also propose a change monitoring and notification system *WebSCAN*(Web Sites Change Analyzer and Notifier), which monitors and analyzes the changes of the Web to notify a user the important changes by a push-type delivery mechanism.

2.2 Related Work

Allan[James Allan, 1998] and Yang[Yiming Yang, 1998] studied on the topic detection and tracking (TDT) issues. They developed some algorithms for discovering and threading together topically related material in streams of data such as newswire and broadcast news. In contrast, we propose some new criteria to find valuable information, such as fresh, popular and urgency information, from the data stream.

Munakata[Munakata *et al.*, 2000] proposed some criteria for choosing appropriate data from periodically generated data sequences: freshness and synchronousness of the data. In their work, freshness is defined as the time duration between the current-time and the oldest data's time-stamp. On the other hand, synchronousness is defined as the time duration between the newest data and the oldest data. In contrast, we consider the freshness of the data from both time and content perspectives.

PointCast[PointCast Network, 1999][Ramakrishnan and Dayal, 1998] is a dissemination service that has attracted a large population of users. It obtains profiles from users in which they subscribed their interest channels (sub-channels), and uses these profiles to assemble and update customized "newspaper" from a database of current stories. As the customizing of a user is limited to add (remove) the channel (sub-channel), the user interests would not be well represented. In other words, the PointCast is a more popular information service than personal one.

On the other hand, SIFT[Yan and Garcia-Molina, 1995], a tool developed for wide-area information dissemination at Stanford University, combines data management ideas from information retrieval with a publish/subscribe model for dissemination. The approach taken by SIFT, requires users to explicitly submit their profiles and updates those profiles using relevance feedback. SIFT can answer for the complex interests of a user well, but the omission of information status, update frequency, dissimilarity etc., may raise a problem that misses some user needs, both interesting and importance information.

[Shapiro, 1998] has developed the techniques that can create "channels" by dynamically filtering the large-scale web content. The goal of [Shapiro, 1998] is to change the way of presenting information to users from the traditional pull-based browsing model to a push-based channel paradigm. However, it also ignores the information status, and creates channels directly form

results without restructuring.

The ANATAGONOMI[Kamba *et al.*, 1997] is a novel push-type news delivery system that uses the user behavior as a feedback to make an automatic presentation of user interesting information. The key idea of [Kamba *et al.*, 1997] is that the user behaviors can be accumulated as user profile and present the relevant information for user automatically. Another work, which has been done at Incubation Center of NEC corporation, developed a theater-style web browsing system *SiteCruise Theater*[Theater, 1997]. In [Theater, 1997], web pages are classified by category or theme and provided just like a movie or TV program automatically.

Some works for the change detection over wrapped web pages have been done at C3 project[C3Project, 2001]. One of the main contributions of C3 project is to portray the changes between two structured data in a succinct and descriptive way: *meaningful change detection*[Chawathe and Garcia-Molina,]. They also consider the data structure to detect the changes. At the contrast, we are interested in estimating web changes using their content and structure to pick up the valuable information, rather than change detection.

Netmind[NetMind, 2001] is a typical URL changes monitor system that extends the Web search engines. Netmind also notifies users the change information using the push technology. WebGUIDE[Douglis *et al.*, 1996] is another system for exploring changes to web pages and web structure that supports recursive document comparison. The contribution of WebGUIDE is to support recursive document comparison and a difference viewing by a graphical navigator. However, the change semantics, such as freshness and popularity, are not considered in these conventional systems. In the nutshell, these systems just detect the changes, but not discover the important ones from massive changes.

WebCQ[Liu *et al.*, 2000] is a system that discovers and detects the changes of the web pages, and notifies user of interesting changes with personalized customization. Features of WebCQ include the capabilities for monitoring and tracking various types of changes, personalized delivery of page change notifications and personalized summarization of changes. However, as same as other conventional systems, the worth of web page change is not considered at WebCQ. In addition, the notification is based on the user interests. This feature makes it necessary to specify user interests clearly. Since incoming information is not foreseeable and the Web is changed continuously, it's not easy to specify user profile to acquire the new valuable information.

2.3 Time-series Features

2.3.1 Retrospective Scope

A retrospective scope is a collection of articles, web pages or page fragments to compare with the incoming or changed one for computing the time-series features, such as freshness and popularity.

Basically, we can collect the articles or web pages which are delivered or changed during a fixed-duration to construct the retrospective scope. For example, for a new article a, we can select the articles during past three months as the retrospective scope used to compute the time-series features of a.

We also can fix the number of articles contained in the retrospective scope to construct the

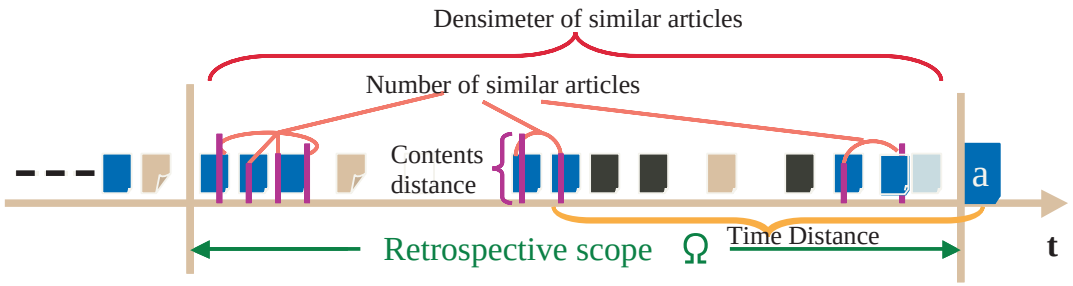

Figure 2.1: Freshness

retrospective scope. For example, we can collect the past 1000 articles as the retrospective scope for a new article.

Moreover, we can fix the number of its similar articles delivered before. For example, we can construct the retrospective scope by restricting the number of the similar articles of the new one to be 100.

2.3.2 Freshness

The articles (or web pages), which are quite different from previous articles, would be valuable. In other words, we can say that the articles have their freshness and uniqueness. Indeed in some cases, these articles may be *scoop* news.

As shown in **Figure** 2.1, the freshness of the article a can be computed by

- the number of its similar articles in a retrospective scope, denoted by $fresh_{num}(a)$,
- the dissimilarity between a and the past articles in a retrospective scope, denoted by $fresh_{cd}(a)$,
- the density of its similar articles in a retrospective scope, denoted by $fresh_{de}(a)$, and
- the time distance of a and its similar articles in a retrospective scope, denoted by $fresh_{td}(a,\omega)$.

The integrated freshness of an article a compared with articles in a retrospective scope Ω, denoted by $fresh_{\Omega}(a)$, is also defined as follows:

$$\begin{aligned} fresh_{\Omega}(a) = \quad & \alpha * fresh_{num}(a) \\ & +\beta * fresh_{cd}(a,\omega) \\ & +\gamma * fresh_{de}(a) \\ & +\sigma * fresh_{td}(a,\omega) \end{aligned} \tag{2.1}$$

where $\alpha,\beta,\gamma,\sigma$ are the weight values. ω is the set of a's similar articles in the scope Ω.

Let m and n be the numbers of articles in ω and Ω, respectively. The above four types of freshness measurements are defined as follows.

Freshness based on the number of similar articles When there are few articles that are similar to a in Ω, we can say a is newer one and its freshness is considered to be high. So, we have

$$fresh_{num}(a) = \frac{1}{log_2(2+m)} \tag{2.2}$$

The similarity $sim(a,b)$ of two articles a and b are computed based on vector space model[SALTON, 1968] as follows.

$$sim(a,b) = \frac{v(a) \cdot v(b)}{|v(a)||v(b)|} \tag{2.3}$$

where, $v(a)$ and $v(b)$ are keyword vectors of a and b, respectively.

If the similarity of a and b is greater than a pre-specified threshold, we say a and b are similar.

Freshness based on the content distance The *content distance* of article a and b can be defined as follows:

$$dis(a,b) = |v(a) - v(b)| \tag{2.4}$$

where, $v(a)$ and $v(b)$ are keyword vectors of article a and b, respectively.

The content distance can represent, comparing with b, how much new information has been added into a. Therefore, we can say, the content distance between a and its similar articles is bigger, the freshness of a is higher. Thus, we have

$$fresh_{cd}(a,\omega) = \log(\frac{1}{m} \sum_{i=1,b_i \in \omega}^{m} dis(a,b_i)) \tag{2.5}$$

Freshness based on the density of similar articles The *density* d of the similar articles of a in Ω is computed as m/n. Here, d can be considered as the appearance probability of a in Ω. When d is small, a is rare one, and its freshness is considered to be high. So, we have

$$fresh_{de}(a) = log_2 \frac{n}{m} \tag{2.6}$$

Freshness based on the time distance Assume that some articles in the past archive are similar to article a and that the time distance between a and those similar articles is large. Intuitively, in this case, some new information is considered to occur. Thus, the article a is considered to have a high freshness. So, we have

$$fresh_{td}(a,\omega) = \log(\frac{1}{m} \sum_{i=1,b_i \in \omega}^{m} (t(a) - t(b_i))) \tag{2.7}$$

where $t(a)$ is time stamp of a.

2.3.3 Popularity

The articles, which are quite similar to almost of the previous ones, would be also valuable. For example, when an incident happened, a series of reporting articles would be sent continuously. The articles described one of the *hottest* topics at that time. That is, they were popular.

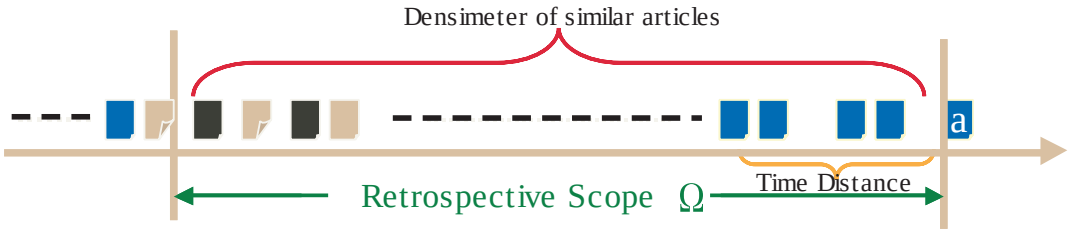

Figure 2.2: Popularity

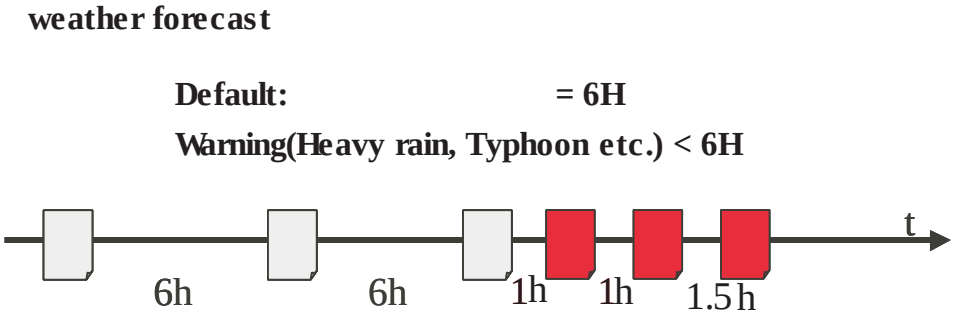

Figure 2.3: Example of Update Frequency

The popularity of the article a can be computed by 1) the density of its similar articles of a in a retrospective scope, and 2) the time distance of a and its similar articles in a retrospective scope (see **Figure** 2.2). In other words, if a has many similar articles in a retrospective scope and the time distance among them is small, the popularity of a is considered to be high. Thus, we have

$$pop(a) = e^{\lambda_1 k} + e^{-\lambda_2 t_d} \tag{2.8}$$

where $\lambda_1 (> 0), \lambda_2 (> 0)$ are the weight values. $k = m/n$ is the density of similar articles. t_d is the time distance of a and its similar articles, defined as follows:

$$t_d = \frac{1}{m} \sum_{i=1, b_i \in \omega}^{m} (t(a) - t(b_i)) \tag{2.9}$$

2.3.4 Urgency

Each channel generally has its default update duration. However, in some cases, the duration becomes shorter than the default one because of urgency. For example, the channel of weather forecast would change their update frequency when typhoon warnings were announced (see **Figure** 2.3). In this case, the update duration has been changed from six hours to one hour.

The urgency of a belonging to channel c is computed as follows:

$$freq(a) = e^{\lambda_1 \sigma_c} \tag{2.10}$$

$$\sigma_c = \frac{D_c - d_c}{d_c} \tag{2.11}$$

where, D_c is the default update duration of channel c. d_c is the latest update duration of channel c. λ_1 is a weight value.

2.4 Filtering Based on User Profile and Temporal Criteria

Based on the user profile and the time-series features, we give a new filtering model for the incoming articles and new web pages. In this model, the filter has three functions: (a) user profile matching, (b) channel (web site) update frequency monitoring, and (c) popularity and freshness calculating. When the user profile for the filtered channel is q, the score of article a via channel c is calculated by following equation:

$$\begin{aligned} score(a) &= \alpha' * sim(a,q) + \beta' * freq(a) \\ &+ \gamma' * (max(\mu * pop(a), \nu * fresh(a)) \end{aligned} \tag{2.12}$$

where $sim(a,q)$ is the similarity between article a and user profile q based on vector space model[SALTON, 1968]. $freq(a)$ is the urgency of a belonging to channel c. $pop(a)$ is the popularity of a. $fresh(a)$ is the freshness of article a. α', β', γ', μ, and ν are weight values for each term.

The articles which have higher score (greater than a pre-defined threshold), will be selected as the valuable ones and presented to the user.

2.5 Evaluation

To evaluate the temporal criteria proposed in this thesis, we conducted the following realistic tests:

- evaluation of effects of retrospective scope on temporal criteria, and
- evaluation of temporal criteria based filtering model.

2.5.1 Evaluation of Effects of Retrospective Scope on Temporal Criteria

To compute the temporal criteria (time-series features), we need to construct the retrospective scope at first. Different retrospective scopes may bring up different values. Moreover, some systems require a small size of retrospective scope for online processing. It is not easy to specify a proper retrospective scope. Luckily, as the evaluation results shown, if we select proper parameters to compute the integrated freshness and popularity, the effects of retrospective scope on them can be reduced. In this evaluation, we used 1014 articles collected from CNN channel of PointCast Network[PointCast Network, 1999].

Freshness

(a) Retrospective Scope Constructed by the Total Number of Articles In this evaluation, as shown in **Table** 2.1, we conducted three evaluations fixing the total number of articles contained in retrospective scope to be 100, 200 and 500, respectively.

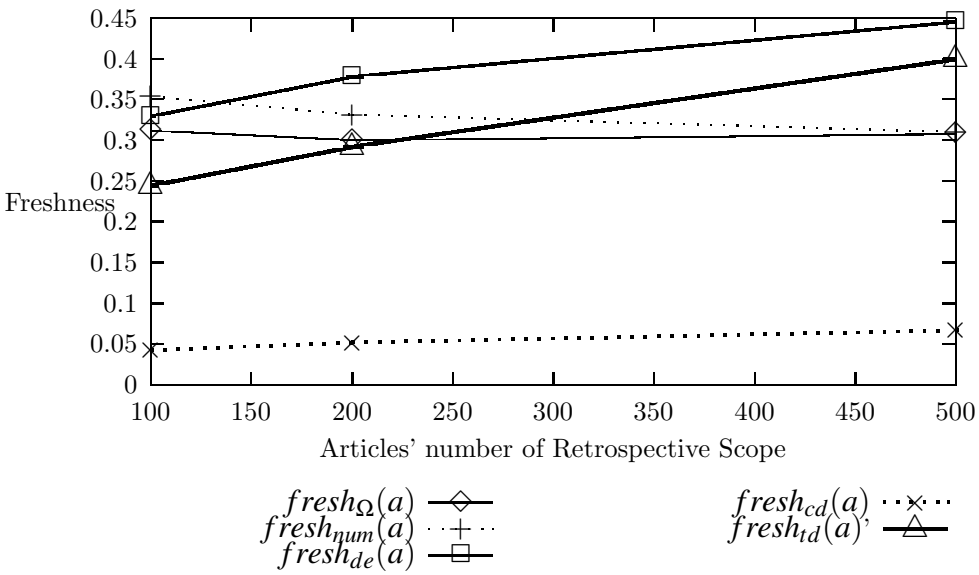

Figure 2.4: Freshness and Retrospective Scope Based on Total Number of Articles

Table 2.1: Freshness and Retrospective Scope Based on Total Number of Articles

Times	Total Number of Articles	Number of Similar Articles	Time Interval	Integrated Freshness
First	100	7.90	58.71	0.311
Second	200	11.22	115.13	0.303
Third	500	19.27	313.11	0.306

In the first evaluation, the number of articles within the retrospective scope was 100. Within the retrospective scope, the average number of similar articles of each estimated one was 7.90. The average time interval was 58.71 hours. The average integrated freshness was estimated to be 0.311*.

In the second evaluation, the number of articles within the retrospective scope was 200. The average number of similar articles and time interval were 11.22 and 115.13 hours, respectively. The average of integrated freshness was 0.303.

In the third evaluation, the number of articles contained in retrospective scope was 500. The average number of similar articles and time interval were 19.27 and 313.11 hours, respectively. The average integrated freshness was estimated to be 0.306.

Figure 2.4 shows the relationships between each kind of freshness and the retrospective scope. The correlation coefficients of retrospective scope and each kind of freshness (integrated freshness, freshness based on number of similar articles, freshness based on density of similar

*We formalized the freshness ranging from 0 to 1.

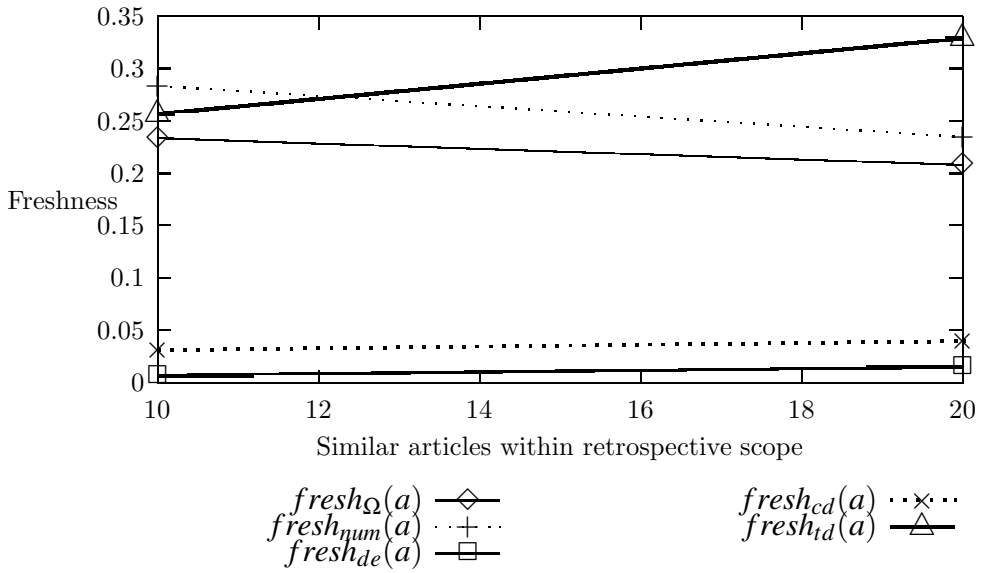

Figure 2.5: Freshness and Retrospective Scope Based on Number of Similar Articles

Table 2.2: Freshness and Retrospective Scope Based on Number of Similar Articles

Times	Total Number of Articles	Number of Similar Articles	Time Interval	Integrated Freshness
First	92.63	9.55	49.41	0.229
Second	155.28	18.26	78.81	0.205

articles, freshness based on content distance and freshness based on time distance) had been estimated to be -0.559, -0.996, 0.999, 0.998 and 0.991, respectively. It is obvious that the coefficient correlations of retrospective and each kind of freshness are high. With the growth of retrospective scope, the factors, density of similar articles, content distance and time distance, would increase the freshness value. On the other hand, the number of similar articles would decrease the value of freshness. The density of similar articles was the most effective factor. Integrated freshness could mitigate the effects of retrospective.

(b) Retrospective Scope Constructed by the Number of Similar Articles In this evaluation, we constructed retrospective scope by fixing the number of similar articles to be 10 and 20 for two tests. As shown in **Table** 2.2, in the first evaluation, the average number of articles contained in retrospective was 92.63. The average number of similar articles and the average time interval were 9.55 and 49.41 hours, respectively. The average integrated freshness was estimated to be 0.229.

In the second evaluation, the average number of articles contained in retrospective was 155.28. The average number of similar articles and the average time interval were 18.26 and

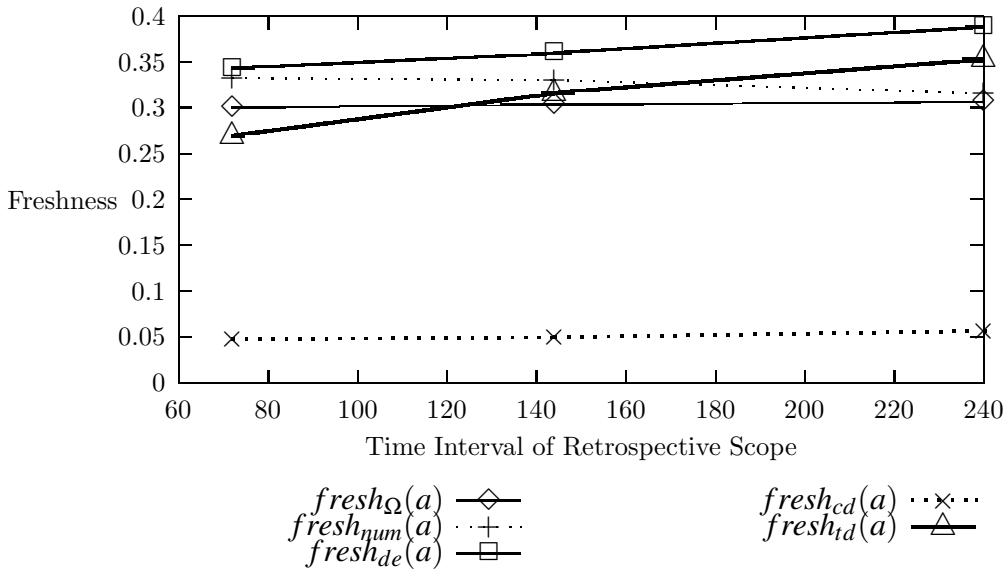

Figure 2.6: Freshness and Retrospective Scope Based on Time Interval

Table 2.3: Freshness and Retrospective Scope Based on Time Interval

Times	Number of Total Articles	Number of Similar Articles	Time Interval	Integrated Freshness
First	169.32	10.68	80.13	0.304
Second	221.21	12.84	170.05	0.306
Third	272.60	13.81	252.01	0.307

78.81 hours, respectively. The average integrated freshness was estimated to be 0.205. The average numbers of similar articles were less than specified values (10 and 20), because that some articles had fewer similar articles in the test data collection.

As shown in **Figure** 2.5, time distance, density of similar articles and content distance would increase the freshness value with the growth of retrospective scope. The number of similar articles would decrease the value of freshness. At this time, time distance was the most effective factor. Integrated freshness was decreased slowly by the change of retrospective scope.

(c) Retrospective Scope Constructed by Time Interval In this evaluation, we constructed retrospective scope based on time interval. We have configured the time interval to be 72 hours, 144 hours and 240 hours for three tests, respectively. As shown in **Table** 2.3, in first evaluation, the average number of articles contained in retrospective scope was 169.32. The average number of similar articles was 10.68 and the average time interval was 80.13 hours. The average value of integrated freshness was estimated to be 0.304.

In the second evaluation, the average number of articles and similar articles were 221.21 and

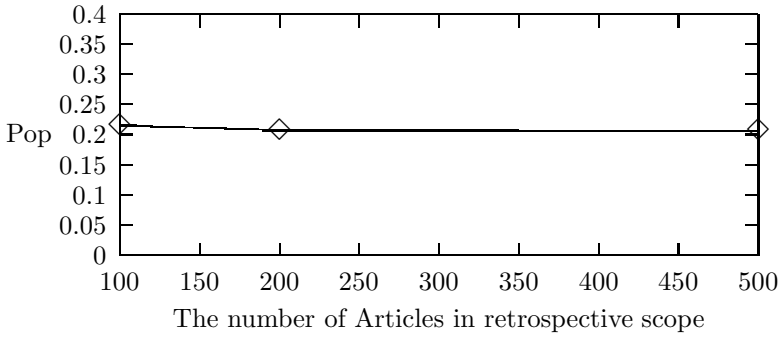

Figure 2.7: Popularity and Retrospective Scope Based on Total Number of Articles

12.84, respectively. The average time interval was 170.05 hours. The average value of integrated freshness was estimated to be 0.306.

In the third evaluation, the average number of articles contained in retrospective scope and similar articles were 272.60 and 13.81, respectively. The average time interval was 252.01 hours. The average value of integrated freshness was 0.307. The average time intervals were different from the configured values because the delivery time of news articles in PointCast was unfixed.

As shown in **Figure** 2.6, the correlation coefficients of retrospective scope and each kind of freshness (integrated freshness, freshness based on number of similar articles, freshness based on density of similar articles, freshness based on content distance and freshness based on time distance) had been estimated to be 0.980, -0.947, 0.997, 0.984 and 0.988, respectively. The most effective factor was density of similar articles. The results show that the integrated freshness could reduce the effect of retrospective scope on it.

Popularity

As same as freshness, the computation of popularity is also depended on retrospective. **Figure** 2.7, **Figure** 2.8, and **Figure** 2.9 show the evaluation results of popularity by constructing the retrospective scope based on the number of articles, the number of similar articles and time interval, respectively. From these results, we can say, the retrospective scope has few effects on the computation of popularity proposed in this thesis.

2.5.2 Evaluation of Temporal Criteria Based Filtering Model

We evaluated the temporal criteria based filtering model by comparing it with the conventional one based on user profile. As shown in **Figure** 2.10, we built two filters in this evaluation. Filter 1 was the conventional one based on the matching of user profile and the incoming articles. Filter 2 used both user profile and temporal criteria to select valuable information.

In this evaluation, we used the news articles delivered in PointCast during one week as the test data set. The test data set was composed of 421 news articles of CNN channel, 135 news articles of ZDNet channel and 512 news articles of Sports channel. The size of retrospective

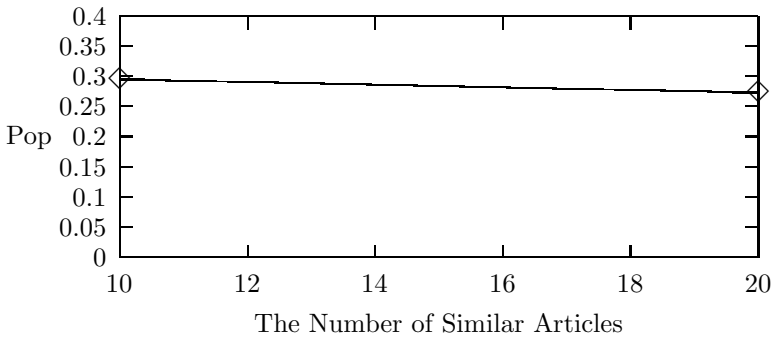

Figure 2.8: Popularity and Retrospective Scope Based on Number of Similar Articles

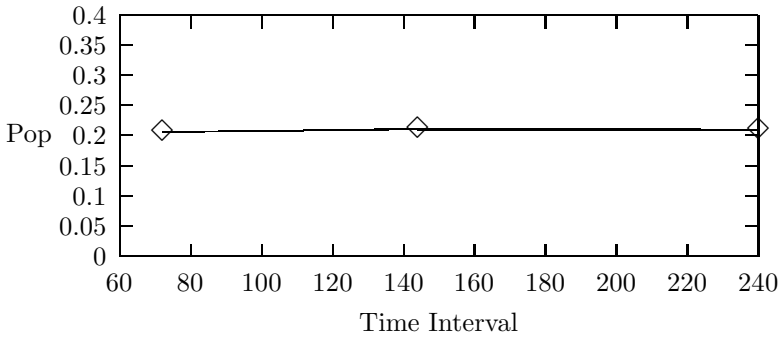

Figure 2.9: Popularity and Retrospective Scope Based on Time Interval

scope for computing temporal criteria was 150. That is, we used 150 past articles for each new article to compute its temporal criteria. For filter 1, the threshold of similarity between user profile and new article had been set to be 0.65 based on some preliminary tests. Filter 2 used the function 2.12 to select valuable news articles: $fresh(a)$ was computed by the integrated freshness $fresh_{\Omega}(a)$; $\alpha, \beta, \gamma, \delta$ were set to be $0.7, 0.1, 0.1, 0.1$, respectively; $\alpha', \beta', \gamma', \mu, \nu$ were

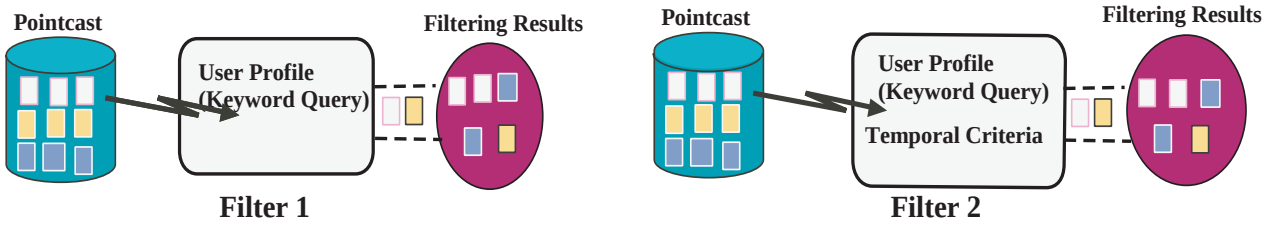

Figure 2.10: Evaluation Environment of Comparing Experiment

Table 2.4: Results of Comparing Evaluation

	All Articles			Results Set of Filter 1			Results Set of Filter 2		
	Min.	Max.	Average	Min.	Max.	Average	Min.	Max.	Average
Similarity	0	0.810	0.341	0.640	0.810	0.711	0	0.810	0.635
Urgency	0.606	54.5	1.79	-	-	-	0.606	54.5	1.95
Integrated Freshness	0.354	1.0	0.673	-	-	-	0.462	1	0.717
Popularity	0	0.865	0.235	-	-	-	0	0.865	0.386

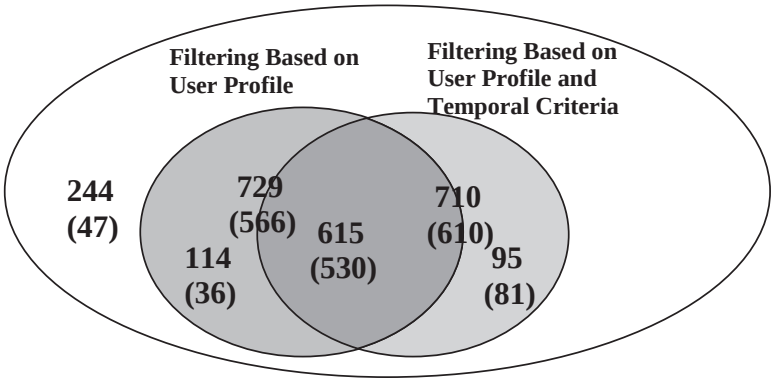

Figure 2.11: Article Sets: Total, Results of Filter 1 and Filter 2. ($x(y)$ means the number(x) of articles of each set, and the number(y) of relevant articles.)

set to be $5, 1, 4, 1, 1$, respectively; the threshold was 4.0. These two filters used same user profile containing the following keywords: *sports, basketball, football, baseball, volleyball, NBA, NHL, MLB, soccer, game, match, Lakers, world, national, team, IT, hardware, software, application, UNIX, Linux, windows, OS, Y2K, virus, database, internet, XML, java, mobile.*

Table 2.4 shows the evaluation results. Where, similarity, integrated freshness and popularity were formalized to range from 0 to 1. The default value of urgency was 1.0.

As shown in **Table** 2.4, filter 2 had selected the articles which had the minimum value of similarity, urgency and popularity. One considerable reason is that filter 2 selected articles based on the sum of values of all criteria. 58 articles were selected although their similarity values were 0. Their average value of urgency, integrated freshness and popularity were 2.45, 0.85 and 0.434, respectively. In addition, about half of them (28 of 58) were the articles processed at the starting time point. Because they had no past articles, their freshness values were very high and then they had been selected by filter 2.

As shown in **Figure** 2.11, filter 1 selected 729 articles from 1068 articles. Filter 2 selected 710 articles. 615 articles were selected by both filter 1 and filter 2. Because the news articles had been clustered into categories previously in PointCast, filter 1 could select many articles. The recall and precision ratios of filter 1 were 0.816 and 0.776, respectively. On the other hand, the recall and precision ratios of filter 2 were 0.881 and 0.861, respectively. Here, the recall ratio is

the rate of relevant articles selected by filter 1 (or filter 2) among all relevant articles. [†] Precision ratio is the rate of relevant articles selected by filter 1 (or filter 2) among all it selected articles. Only from the recall and precision ratios, we may not be able to say filter 2 is better than filter 1. But at least, we can say, filter 2 can select fresh, popular and urgent information which could not be selected by filter 1 from the incoming information.

114 articles were selected by filter 1 while they were not selected by filter 2. Their average values of similarity, integrated freshness and urgency were 0.752, 0.544 and 0.915, respectively. Among them, 36 articles were missing articles which were not selected by filter 2 although they were relevant articles.

95 articles were selected by filter 2 while they were not selected by filter 1. Their average values of similarity, integrated freshness and popularity were 0.224, 0.839 and 38.251, respectively. Among them, the number of failure articles which were not relevant articles but selected by filter 2 is 14.

Ideally, the missing ratio (r_m) and failure ratio (r_f) defined as follows are 0. In our evaluation, the missing and failure ratios were 0.316 and 0.147, respectively. In other words, comparing to filtering method based on user profile only, the filtering method by using both user profile and temporal criteria could select additional information which are fresh/popular/urgent with the failure ratio 14.7% and missing ratio 31.6%.

Missing ratio r_m and failure ratio r_f are defined as follows.

$$r_m = \frac{|M|}{|T - T \cap V|} \tag{2.13}$$

$$r_f = \frac{|F|}{|V - V \cap T|} \tag{2.14}$$

where, M and F are the sets of missing articles and failure articles, respectively. T and V are the article sets selected by filter 1 and filter 2, respectively. $|X|$ stands for size of article set X.

2.6 Application System: WebSCAN

Based on the temporal criteria, we propose a change monitoring and notification system *WebSCAN* (Web Sites Change Analyzer and Notifier), which monitors and analyzes the changes of the Web to notify a user the important changes by a push-type delivery mechanism. In WebSCAN, the changes of web sites are monitored periodically. The detected change is estimated by its content, browsing frequency and update frequency. That is to say, we estimate the worth of a change by computing its freshness, popularity and urgency. The retrospective scope is constructed based on structure of web page and web site. Based on the estimated change worth, the important changes will be selected and delivered to users automatically with the push technology.

In contrast to earlier works concerned with the *Web change notification*, the main contributions of WebSCAN proposed in this thesis can be summarized as follows:

- **Content and Structure-based Change Analysis**
  To discover the higher worth information from the changed web pages, the change worth

[†]The relevant articles were selected by a user with considering: weather they matched up the specified keywords; weather they described fresh or popular or urgency information.

is computed by both content-based and structural approaches. The former is based on computing the similarity/dissimilarity between newly added content and previous content. The latter is to consider the browsing frequency, update frequency, and the place of the changed web page.

- **Semantics of Change Information: Freshness and Popularity**
  We compare the changed pages with the previous pages and compute their similarities/dissimilarities to evaluate the change worth: If the change is not similar to previous pages, it will bring the *fresh* information. On the other hand, the new page that is similar to previous pages may bring *popular* information.

- **Push-type Change Notification with Personalization**
  One of the efficient ways to obtain new information is push technology[Aksoy *et al.*, 1998]. In our approach, based on the change worth, the notification, which contains the selected change's information, is generated and delivered to the users automatically. Each user can use his/her own profile to filter and view the received notification in his/her original way.

2.6.1 Change Analysis

Comparison Scope

A *comparison scope* is a collection of web pages or page fragments to compare with the change for computing the change worth. Actually, the comparison scope is the retrospective scope for computing time-series features of changes. Each member of a comparison scope has some relation to the change: similar, same topic, former version and so on. The web changes have variant type, such as update, adding new page and so on. According to the change type, it's necessary to select the proper comparison scope (paragraphs, pages and so on.) to compute the freshness and popularity of the change.

We choose the members of comparison scope based on the web structure (analyzed by URL path) or the page structure (analyzed by Document Object Model). In most web sites, the related pages are organized into one directory. Thus, a directory can be roughly regarded as a certain topic. Here, the directories of a web site are analyzed based on URL paths. For example, the directory *foo* of web site *www.foo.com* means the URL *http://www.foo.com/foo/*. With this assumption, the members of the comparison scope are selected as follows: at first, we represent a web site as a tree based on analyzing its URL paths. Secondly, we select all of the change's siblings as the members of its comparison scope. Meanwhile, when a paragraph is added to an existing page (page modification), the previous existed paragraphs are collected as the members of its comparison scope.

(a) Page Modification In the case of page modification, as shown in **Figure** 2.12, at first, we partition the modified page into some units at the level as same as the change. These partitioned units are then collected into the comparison scope to compute the change worth.

(b) New Page In the case of new page addition, the comparison scope is a collection containing all the siblings of the new page. Moreover, the descendants of its sibling are also contained in the

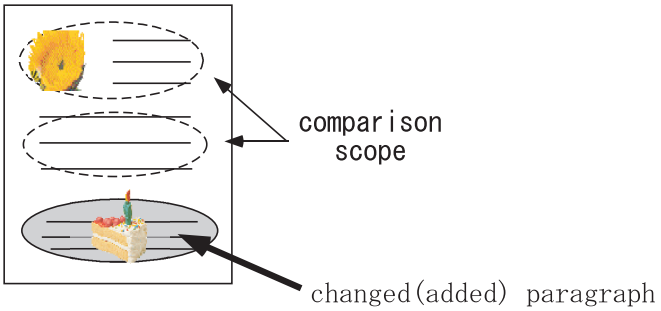

Figure 2.12: Comparison Scope: Case of Page Modification

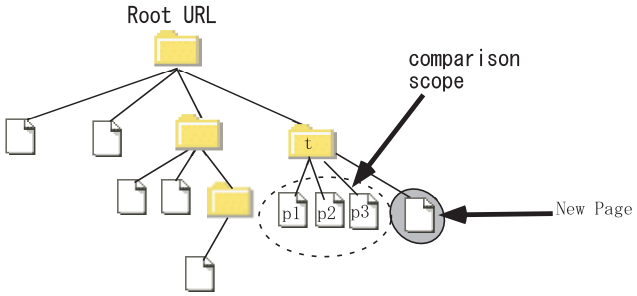

Figure 2.13: Comparison Scope: Case of New Page

comparison scope. For example, as shown in **Figure** 2.13, the comparison scope is composed of page $p1$, $p2$ and $p3$. ‡

(c) New Topic When a new topic (directory that contains some new pages) has been added, we can regard the added topic as a "virtual page" to select the comparison scope as same as that a new page is added. For instance, as shown in **Figure** 2.14, the new topic st_{new} will be compared with the children ($p2$ and $p3$) of st_{old} and page $p1$ to compute its change worth.

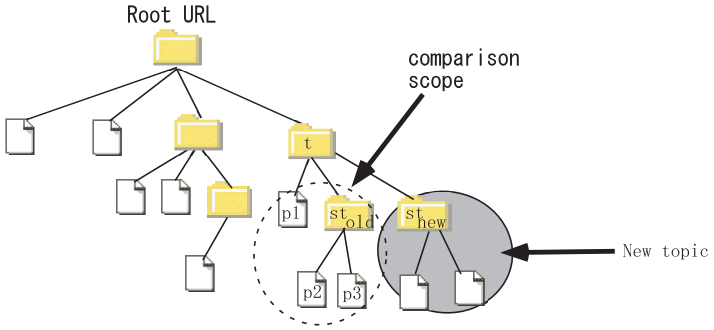

Figure 2.14: Comparison Scope: Case of New Topic

‡Hereafter, as shown in **Figure** 2.13, a web site is represented as a tree based on the URL path analysis and each edge means the directory path.

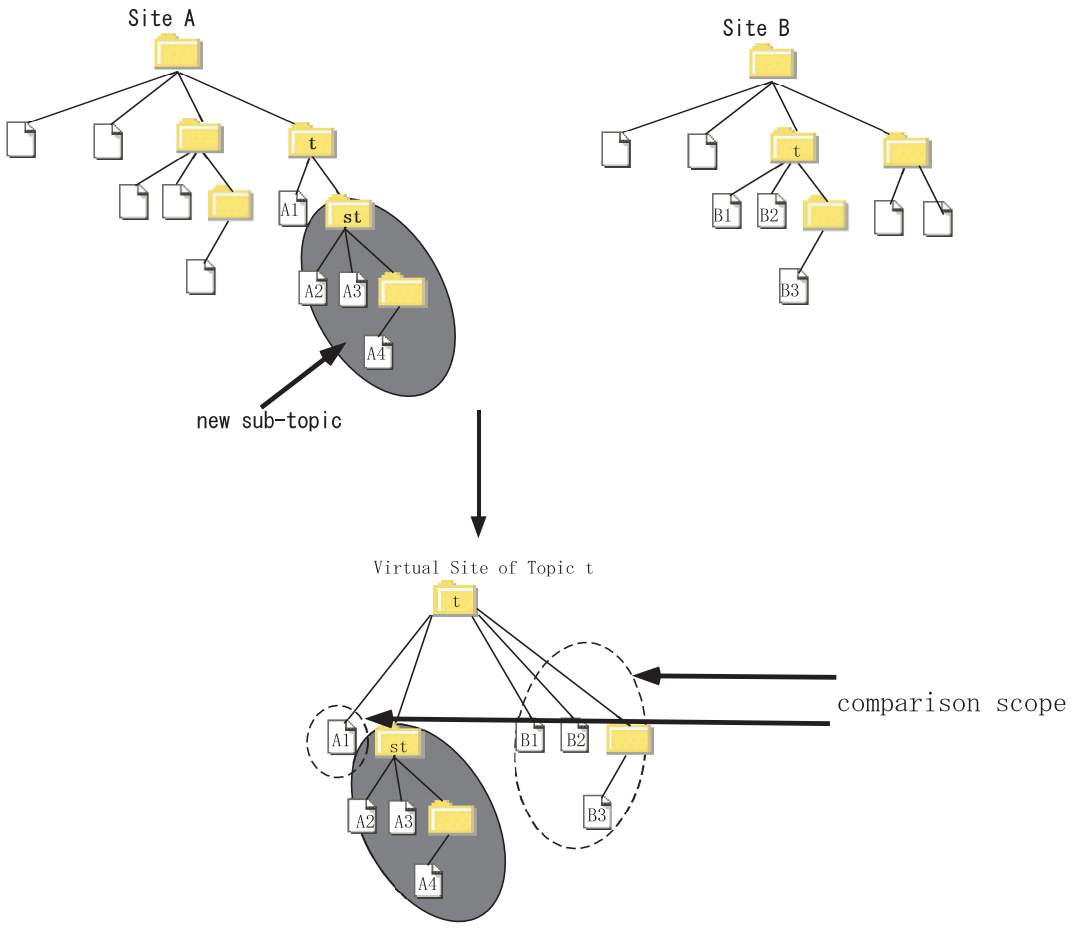

Figure 2.15: Comparison Scope: Case of Related Web Sites

(d) Related Web Sites Since many web sites deliver the similar information, the correlation of them should not be overlooked during the change analysis. Since these related sites have high similarities, it's possible to reorganize them into a virtual web site per topic. As shown in **Figure** 2.15, our idea is to organize the related directories of different web sites to one new virtual directory per topic. After that, we can select the members of comparison scope as same as we doing at a single site.

Estimation of Change Worth

In WebSCAN, change worth is estimated by freshness, popularity, and urgency of the change. For simplicity, hereafter, we assume that the detected change is the new page. In the other change case, such as new topic, related web sites and so on, the change worth can be estimated in the same way.

We can compute the freshness, popularity and urgency of a changed page by the definition described above. We should use the comparison scope to compute these features. However, in contrast to a data broadcasting system which updates information periodically, the web sites are changed arbitrarily. Thus, we should modify the computation of the urgency. Our hypothesis is

that, in freshness perspective, the longer the update time interval is, the bigger the change worth is. For instance, when a web site has been updated after a long no-update time, the changes of this site will have high change worthies. At the contrast, in popularity perspective, the shorter the update time interval is, the smaller the change worth is. That's to say, when a web site updates its pages frequently, there maybe some urgency or popular event occurred. Thus, these updated pages are valuable to be notified.

In the freshness perspective, the change worth based on the update time interval of change c (or directory which c belongs to) is defined as follows:

$$V_{uf-fresh}(c,n) = log(ti(c,n)) \tag{2.15}$$

$$ti(c,n) = \frac{t(n)-t(n-1)+ti(c,n-1)\cdot(n-1)}{n} \tag{2.16}$$

where $t(n)$ is the time-stamp of c at the *n-time* update, n is the updated times and $ti(c,n)$ is (average) update time interval of c at *n-time* update.

On the other hand, in the popularity perspective, the change worth based on the update time interval is defined as follows:

$$V_{uf-pop}(c,n) = 1/V_{uf-fresh}(c,n) \tag{2.17}$$

Usually, a web site may have several topics, and posts related pages to the same directory. That's to say, a topic is often organized into one directory. The browsing frequency of each topic (directory) could signify the interest of user to that topic. A higher interesting topic may have higher browsing frequency. Therefore, the topic of higher browsing frequency should be high value to be notified due to higher user interest. The change c's change worth based on the browsing frequency is defined as follows:

$$V_{browsing}(c) = log(bf) \tag{2.18}$$

where, bf is the browsing frequency of the topic including c.

Consequently, based on freshness/popularity, update frequency and browsing frequency, the change worth of change c, $worth(c)$, is defined as an integrated form:

$$worth(c) = \begin{cases} worth_{fresh}(c) & \text{if user prefers to fresh information} \\ worth_{pop}(c) & \text{if user prefers to popular information} \end{cases} \tag{2.19}$$

where,

$$worth_{fresh}(c) = \alpha \cdot fresh(c,\Omega) + \beta \cdot V_{browsing}(c) + \gamma \cdot V_{uf-fresh}(c,n) \tag{2.20}$$

$$worth_{pop}(c) = \alpha \cdot pop(c,\Omega) + \beta \cdot V_{browsing}(c) + \gamma \cdot V_{uf-pop}(c,n) \tag{2.21}$$

$$\alpha+\beta+\gamma = 1.0, \alpha > 0, \beta > 0, \gamma > 0$$

where α, β, γ are user definable weight values.

If the change worth $worth(c)$ is bigger than the threshold value, we say c is an important one and notify users of it.

2.6.2 Push-based Change Notification

One of the notable features of push technology is that the same information is delivered to users. In other words, a user is limited to browse information as same as the others. On the other hand, more and more users require personalized information. This is one of the conflicts of popularization and personalization[Acharya *et al.*, 1997].

Our approach is to separate the personalization method from the popularized notification. As same as the typical systems, same notification is delivered to all registered users. When a user received the notification, he (she) can use his/her profile to filter and fetch his/her original notification from the received one. The filtered notification would be translated into an HTML file, whose layout also can be specified by each user. This means that the delivered notification can be viewed in variant ways.

Since the web sites are changed dynamically, notification timing is also important for assisting user to obtain the right information at the right time. WebSCAN has two options to delivery the notification, real-time mode and periodic mode. At the real-time mode, the important changes will be delivered immediately. On the other hands, the notification will be delivered periodically in the periodic mode.

The notification contains the changes since last-time delivery. The added information of each change, such as URL, summary, freshness, popularity, change-worthies based on browsing frequency and update frequency, are also included.

The summary of each change is simply generated from its title, top sentences and the URLs of its images files (if there are some). With the summary, user can gain some pre-knowledge of the changes to easily judge which is valuable for reading or not, than just be notified the fact that there are some changes.

Typical change notification systems usually use the one-to-one push model to deliver change information due to satisfy the user's variant demands. It's necessary to deliver everyone his/her own notification in these systems. Moreover, it's not easy to often modify one's own profile for fetching some different information. Our approach is using the one-to-n model to deliver change notification to registered users. Each user can use his/her own profile, which is maintained by him/her at the client side, to acquire favorite information and view it in his/her favorite way.

The personalization method of WebSCAN means that each user can

- **specify his/her favorite web sites and topics**:
  Usually, a user has his/her interests which are different from others. In WebSCAN, each user can pre-define his/her favorite web sits and topics in his/her profile to fetch his/her interesting changes.

- **compute his/her change worth**:
  In WebSCAN, the change worth is computed at the user's side using **Function (2.19)**. A user then can set the parameters according to his/her interests.

- **define the layout of presentation**:
  Filtered information is translated into an HTML file and presented to user via browser, such

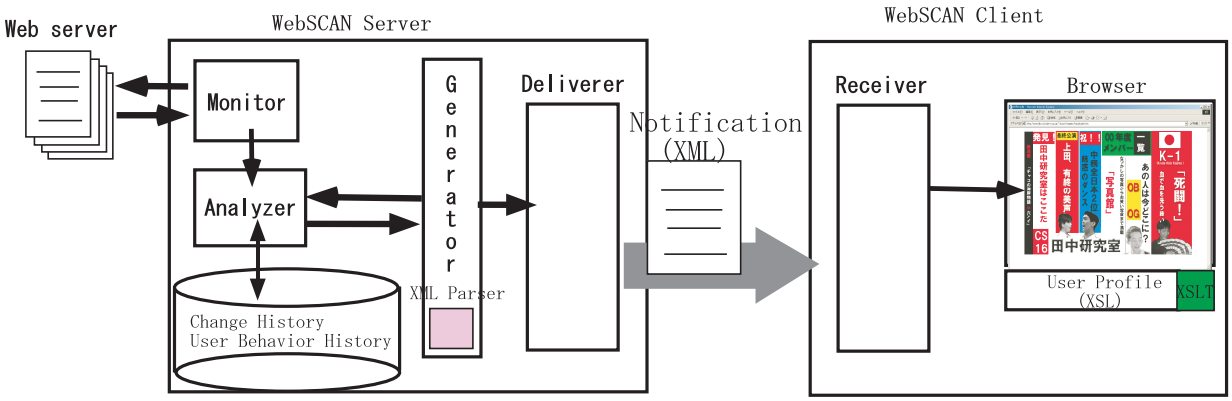

Figure 2.16: Model of Prototype System of WebSCAN

as IE, Netscape and so on. In WebSCAN, a user himself/herself can specify the layout of the outputting HTML file. WebSCAN also provides some default styles, such as hanging-poster like style, newspaper like style and so on.

2.6.3 Prototype System

A prototype system was implemented using Perl and Visual Basic on Windows 2000 platform. The push-type notification mechanism was implemented using the XML/XSLT technology. The XML[Consortium, 1998] formatted notification is generated based on the estimated change worth and delivered to the registered users. Each user uses his/her profile, which is represented as an XSL[XSL, 2001] file, to filter and present the notification.

As illustrated in **Figure** 2.16, the current prototype system has a three-tier structure including (1)*Monitored Web site*, (2) *WebSCAN Server* and (3) *WebSCAN Client*.

WebSCAN server is composed of *Monitor*, *Analyzer*, *Generator* and *Deliverer*. A database of user behavior history and a snapshot of previous web sites are also included.

The monitor watches the time-stamp and size of all pages in a web site. When some changes are detected, monitor will fetch the changed information and invoke the analyzer to analyze the changes. The analyzer compares the detected change with its comparison scope to estimate its change worth. After the analysis, the generator selects the important changes to generate the notification. The deliverer then delivers this notification to the registered users.

The client of WebSCAN is composed of *Receiver* and *Browser*. The receiver is used to receive the notification delivered by the server. The browser filters and presents the received notification by using an XSL file, which represents user profile. **Figure** 2.17 shows part of a sample notification. Part of sample XSL formatted profile is shown in **Figure** 2.18.

```
....

<NOTIFICATION>
<SITE>
  <URL>http://www.usatoday.com</URL>
  <DIR>
    <URL>BASEBALL</URL>
    <PAGE>
     <URL>hphoto.htm</URL>
     <CHANGE-WORTH>
      <FRESHNESS>0.345</FRESHNESS>
      <POPULARITY>0.673</POPULARITY>
     </CHANGE-WORTH>
     <TITLE>USATODAY.com</TITLE>
     <INDEX>Bush to dedicate ...</INDEX>
     <IMG src="memorial.jpg"/>
    </PAGE>
    ....
  </DIR>
</SITE>
...
</NOTIFICATION>
....
```

Figure 2.17: Example of Notification

Table 2.5: Evaluation Results

	Freshness	Popularity
Average	0.450	0.433
Recall Ratio	0.803	0.351
Precision Ratio	0.564	0.540

2.6.4 Evaluation of Change Analysis

In this subsection, we describe a preliminary evaluation of our approach of change worth computation. Since we did not have access to a large crawl of the Web and we did not fully implement the proposed approach, it was not feasible to do the full change worth computations. Instead, we implemented two simplified version of filters: freshness filter and popularity filter. Furthermore, we also adjusted the values of freshness and popularity ranging from 0 to 1.0 in the preliminary evaluation.

- **freshness filter**
  Only the freshness was used to rank the changed pages. The filtering function was defined as follows:

$$\begin{aligned} worth_{fresh}(c) &= fresh(c,\Omega) \\ &= 0.4 \cdot fresh_{sum}(c,\Omega) + 0.4 \cdot fresh_{cd}(c,\Omega) \\ &\quad + 0.1 \cdot fresh_{de}(c,\Omega) + 0.1 \cdot fresh_{td}(c,\Omega) \end{aligned} \tag{2.22}$$

  If change c's worth $worth_{fresh}(c)$ was bigger than the threshold 0.25, it would be selected as the valuable change.

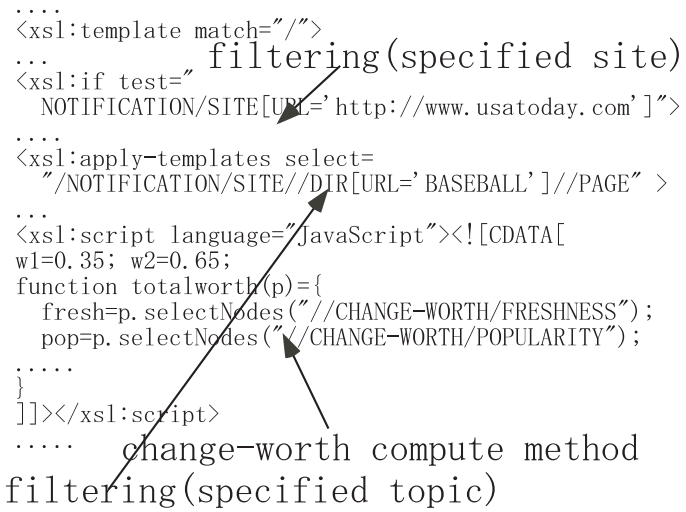

Figure 2.18: Example of User Profile

- **popularity filter**
  Popularity filter used the popularity as the ranking measure. Its filtering function was defined as follows:

$$worth_{pop}(c) = pop(c,\Omega) = 0.5 \cdot e^{0.7d} + 0.5 \cdot e^{-0.3t_d} \tag{2.23}$$

  The threshold for choosing valuable changes was set to be 0.75.

Figure 2.19 and **Figure** 2.20 illustrate the distribution of freshness and popularity based on our evaluation results, respectively. Because that we estimated a small number of changes, some changes had no similar page. In other words, their comparison scopes were empty. In such case, we let its popularity be 0 and let its freshness be 1.0. As shown, excluding these specially cases, the distributions are similar to the regular distribution. Using this feature, we set the thresholds of freshness and popularity to be 0.25 and 0.75 respectively, due to select half of the changed pages as the valuable ones. Based on some early preliminary experiment, the threshold of similarity for deciding the similar pages was set to be 0.6.

Further details of the preliminary evaluation can be summarized as follows:

- One web site, Nikkan Sports (http://www.nikkansports.co.jp), was selected as the monitoring target.
- Web changes were limited to the type of new page addition.
- Two days changes, about 299 pages, were detected from the Nikkan sports site including 6401 pages.

Table 4.2 shows the results of preliminary evaluation. For the freshness, precision ratio was 0.564 and recall ratio was 0.803. On the other hand, the precision and recall ratio for the popularity were 0.540 and 0.351, respectively. Here, the recall ratio is the rate of relevant articles

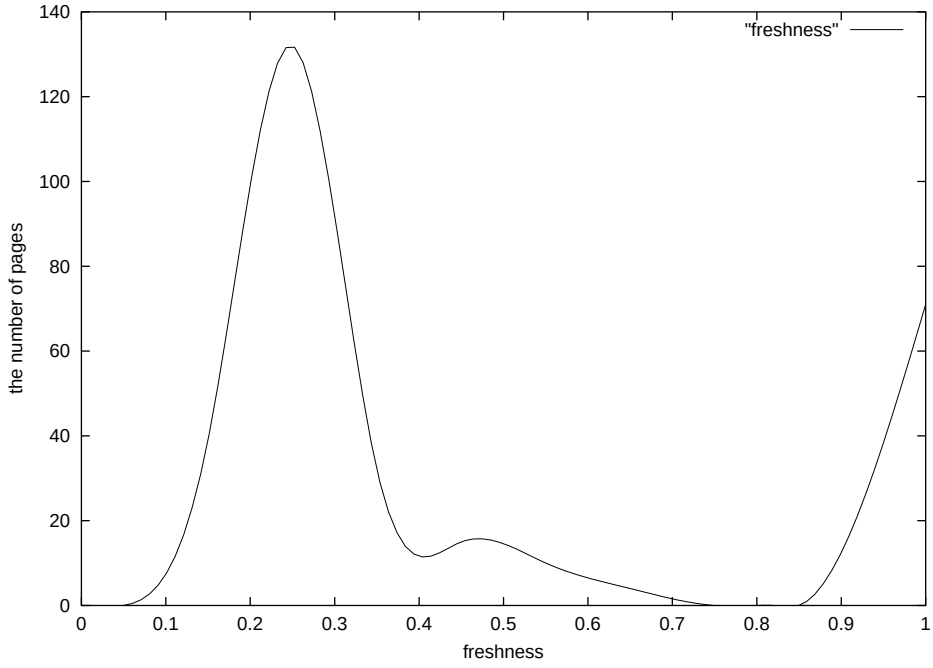

Figure 2.19: Distribution of Freshness: Horizontal axis shows the value of freshness. Vertical axis shows the number of changes.

selected by freshness filter (or popularity filter) among all relevant articles. Precision ratio is the rate of relevant articles selected by freshness filter (or popularity filter) among all it selected articles. In addition, if we compute these ratios excluding the special cases (comparison scope is empty), the precision and recall ratio of freshness filter were 0.65 and 0.718, respectively. The precision and recall ratios of popularity filter were 0.430 and 0.753, respectively. Since our evaluation was a limited one, more improving works need to be done. Nevertheless, these results could confirm that the proposed notions, freshness and popularity, are useful for picking up important information from massive changes.

As mentioned before, the comparison scope is constructed based on the assumption that related pages are organized under same directory. In our preliminary evaluation, in each directory, about 77.6% pages belonged to same topic. Though we estimated only one site, at very least, this shows that the selected one was the kind of comparison scope we are after.

2.7 Application System: Virtual TV Channel

Recent years, broadcast-based (or push-based) information dissemination systems on the Internet are becoming increasingly popular due to advances in the Web technologies. These systems, such as *PointCast*[PointCast Network, 1999][Ramakrishnan and Dayal, 1998], *Backweb*[BAC, 1995], use the push-based technology to disseminate information for users instead of traditional pull browsing paradigm.

The main limitation of the existing broadcast-based information delivery systems is that the information is broadcasted from the standpoint of information providers, and the user interests

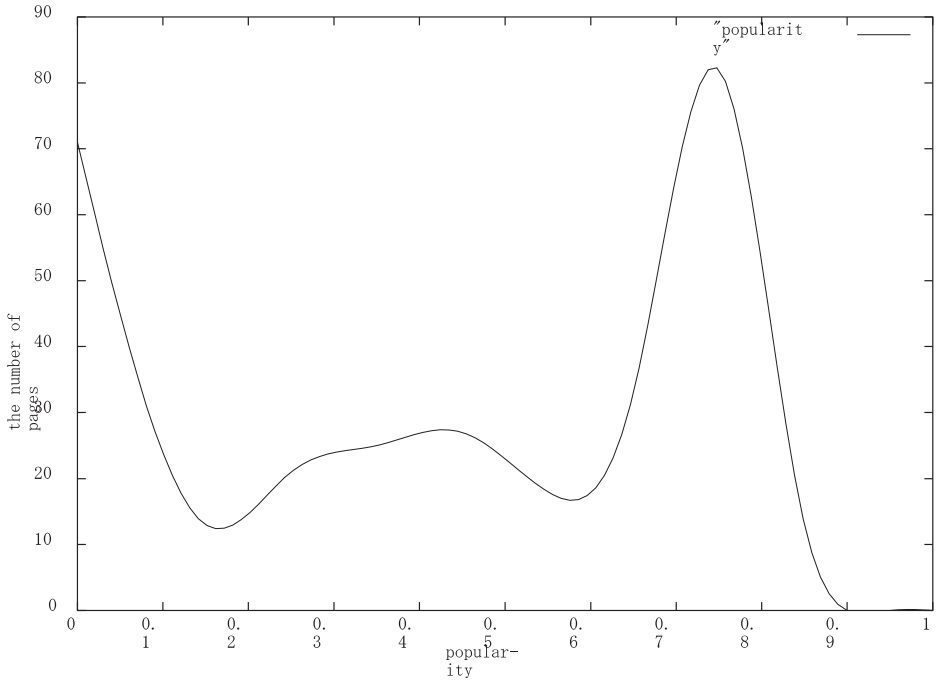

Figure 2.20: Distribution of Popularity (Horizontal axis shows the value of popularity. Vertical axis shows the number of changes.)

are not considered well. This limitation could be summarized as follows:

- **Provider-oriented channel definition**
  With the dramatic increase of the delivered information, the restructuring problem of broadcasted information is becoming very important. In the existing systems, the structure of broadcast information is limitary: they just structure information per channel, and do not analyze the relations among information. So, they could not give a better guide to user where the relevant information is. A user is only allowed to add (remove) a channel or sub-channel as customizing. It is not allowed to select articles by user-defined queries. Furthermore, it is not possible to merge articles from different channels.

- **Filtering by user profiles**
  One of the notable features of push-based, multiple-channel-based information dissemination systems is to send information to users in a form of time-series articles. Conventional information filtering systems does not consider well the *worth* of an article from the standpoint of how many similar articles were previously delivered to users.

- **Browsing interfaces for disseminated information still require user-selection and navigation.**
  Once a user could define his favorite virtual channel, a *couch-potato*-like interface is preferable. But, current push-based viewers do not provide such user interfaces.

We propose a personal on-line news broadcasting system *Virtual TV Channel* based on time-series features of news articles. The virtual TV channel is one broker to filter the information based on time-series features, and to merge them into one virtual channel.

The main features of this system are summarized as follows:

- **Information filtering based on time-series features**
  We propose a new filtering approach that estimate information from both user interests (user profiles) and *information status*. In order to select information from each channel, we estimate not only the similarity between an article and a user profile, but also how often the article's channel is updated (urgency), how fresh or how popular the article is.

- **Analog-like channel definition to merge different channels**
  The selected articles are merged into one virtual channel. In our approaches, it is possible to merge those articles in an *analog manner*. That is, we can specify the merging ration per channel. For example, we can say *merge content of channels X, Y, and Z according to the ration 0.2, 0.3, and 0.5*. In the merging phase, each article is also compared with the previously selected articles. If necessary, the comparison results are fed back to guarantee the given merging ration.

- **TV program-like GUI for viewing virtual channel content**
  The articles of virtual channel are presented like TV-programs. In contrast to typical text-based presentation model, we give a new present model using TVML[TVML, 2003][Hayashi *et al.*, 1999]. That is, based on the feature of content, an appropriate TV-program metaphor (news program, drama, entertainment show, and so on) is automatically selected, and users can enjoy those articles as a TV-program.

As shown in **Figure** 2.21, our *virtual TV channel* consists of *filters*, a *synthesizer*, and a *presenter*. Basically, a user of the virtual TV channel gives the following information:

- His/her interesting real channels
- A user profile per real channel group, and
- A merging ration for filtered channels

For each real channel group, a filter behaves as a broker which filters information of the channel according to a given user profile. A synthesizer merges those filtered information into a virtual channel according to a user-specified merging ration. Then, the information of the virtual channel is presented in the form like TV-program by the *presenter*.

In our system, we mainly handle news articles that are disseminated by push-based channels. So, hereafter, we say *articles* as an information unit of a channel.

In the phase of filtering, the process is done per real channel group. That is, each article in a real channel is compared with the corresponding user-profile, and the similarity of the article with the user-profile is computed. Basically, an article with higher similarity with the user-profile has a high possibility to be selected. We also compute the urgency, freshness and popularity of

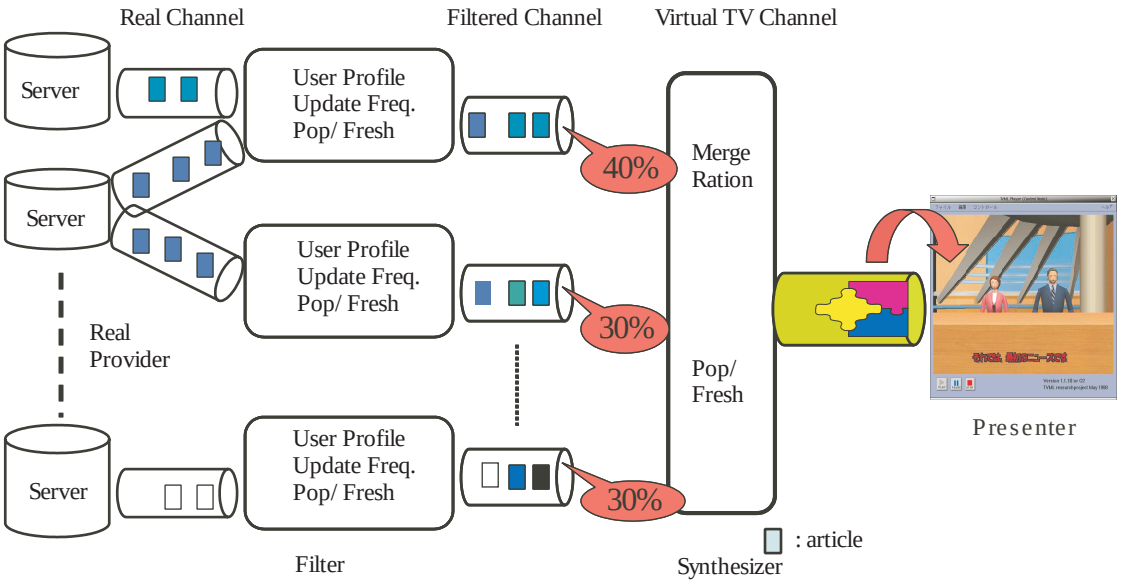

Figure 2.21: Concept of Virtual TV Channel

a new article to select the valuable one. In summary, if the article had high similarity with user profile, high update frequency, and high popularity or freshness, the article would be selected as candidate content of the virtual channel.

After the above filtering process is over, we have multiple series of candidate articles. The synthesizing process is to select articles from those candidates and to merge them into the virtual channel. In this synthesizing process, the system automatically merges the selected articles from each filtered channel by user-specified merging ration. The selection of the articles into final virtual channel is also based on the freshness and popularity of the article with previously selected articles.

Finally, in the presentation phase, the articles of virtual channel are presented in the form like TV-programs. Based on the feature of content, an appropriate TV-program metaphor (news program, drama, entertainment show, and so on) is automatically selected, and users can enjoy those articles as a TV-program.

2.7.1 Filtering Process

Filtered channels consist of articles selected from each real channel group. The filter for each filtered channel has three measures: (a) user profile matching, (b) urgency, (c) popularity and freshness. For each filtered channel, the real channel group and the user profile are specified by a user. When the user profile for the filtered channel is q, the score of article a via channel c is calculated by following equation:

$$fc_{score}(a) = \alpha * sim(a,q) + \beta * freq(a) + \gamma * pop(a) + \zeta fresh(a) \tag{2.24}$$

where $sim(a,q)$ is the similarity between article a and user profile q, $freq(a)$ is the urgency of a belonging to channel c, $pop(a)$ and $fresh(a)$ are the popularity and freshness of article a against

articles previously selected in the filtered channel, respectively. α, β, γ and ζ are weight values for each term.

The similarity of article a and user profile q is computed based on the vector space model[SALTON, 1968].

$$sim(a,q) = \frac{v(a) \cdot v(q)}{|v(a)||v(q)|} \tag{2.25}$$

where, $v(a)$ and $v(q)$ are keyword vectors of a and q, respectively.

2.7.2 Synthesizing Process

After the filtering processes are over, we have some filtered channels. Each filtered channel is a series of candidate articles for virtual channel. The *synthesizer* selects articles from those candidates and merges them into the virtual channel.

In short, the synthesizer computes the score of these candidate articles and selects the high score articles into virtual channel. The score of candidate article a is calculated by following equation:

$$vc_{score}(a) = p_{fc}(t_i) * (w_1 * pop(a) + w_2 * fresh(a)) \tag{2.26}$$

where p_{fc} is the priority of filtered channel which a belongs to. $pop(a)$ and $fresh(a)$ are freshness and popularity of a against previously selected articles in the virtual channel, respectively. w_1 and w_2 are weight values.

In order to guarantee the user specified merging ration, the priority of a filtered channel, i.e., the priority of the all articles in the filtered channel, should be adjusted by the user specified ration and the actual proportion of the filtered channel in the virtual channel. Therefore, the priority of the filtered channel *fc* that the article a belongs to, is calculated by the following formula:

$$p_{fc}(t_i) = \frac{r_{fc}}{ap_{fc}(t_{i-1})} * p_{fc}(t_{i-1}) \tag{2.27}$$

where the r_{fc} is the user specified merging ration of *fc*, $ap_{(t_{i-1})}$ is the actual proportion of *fc*, and the t_{i-1} means the latest iteration.

2.7.3 Content Presentation by TV-program Metaphor

TV-program Metaphors

We have defined five TV-program metaphors to present the articles of virtual channel, where a TV-program metaphor is a pseudo-TV-program described by the TVML. The defined TV-program metaphors are listed in the **Table** 2.6.

TVML (TV-program Making Language)[TVML, 2003][Hayashi *et al.*, 1999], is a way to produce an entire TV-program on the desktop in real time using CG, speech synthesis, and other technology.

Selection of TV-program Metaphor

TV-program metaphors can be manually selected by a user according to his interests. For example, the *Headline* metaphor seems to be suitable to summarize a vast of articles. On the other hand, the *News* metaphor seems to be more suitable for fewer, more important articles.

Table 2.6: TV-program Metaphors

TV-program metaphor	Actions
News	A caster character and a sub-caster character report some articles
Headline	Headline news program style. In this metaphor, a newscaster just summarize the news without giving any comment
Debate	One topic is discussed by some characters
Drama	A drama style program represents a series article
Entertainment show	The articles are reviewed as an entertainment, comic etc.

When each article had high similarity with the other, the *Debate* metaphor seems to be suitable since the presentation can focus on a specific topic.

The basic idea of the way to select an appropriate TV-program metaphor automatically for the articles of the virtual TV channel is to:

- cluster the articles according to their keyword vectors, and then
- analyze the clusters (the number of clusters, the size of each cluster etc.), and
- select an appropriate TV-program metaphor based on the features of clusters.

In order to do the clustering process, we assume that the virtual TV channel has a *buffer*, in which some contiguous articles are pooled, and those articles are presented by a TV-program metaphor.

After the clustering, we can use the status of clusters to select an appropriate TV-program metaphor for these articles. The status of clusters means the number (m), the size ($|C_j|$, the number of the articles in cluster C_j.) and the similarity between the *query vector* and these clusters (S_{C_j}). Here, the *query vector* is either a user profile (keyword vector) or the average of centroid of clusters. The correlations of TV-Program metaphor and the features of clusters are summarized as follows (see also the **Table** 2.7).

- News: the clusters are divided equally and the size of each cluster are generally equal.
- Headline: The clusters are divided equally and the size of each cluster is small.
- Debate: A big cluster is built and the query article is closed to these articles.
- Drama: A big cluster is built and the query article is distance from these articles.
- Entertainment show: When the articles are scattered, an entertainment show, comic and another, can be used to review these various articles.

Table 2.7: Selection of TV-program Metaphor
("-" means this item can be omitted)

TV-program Metaphor	the number of articles (n)	the number of clusters (m)	the size of cluster ($\|C_j\|$)	the similarity of query vector and clusters (S_{C_j})
News	*large*	-	$\simeq \frac{n}{m}$	-
Headline	*small*	-	$\simeq 1$	-
Debate	-	$\simeq 1$	$\simeq$ n	*high*
Drama	-	1	n	*low*
Entertainment show	-	*large*	*large*	-

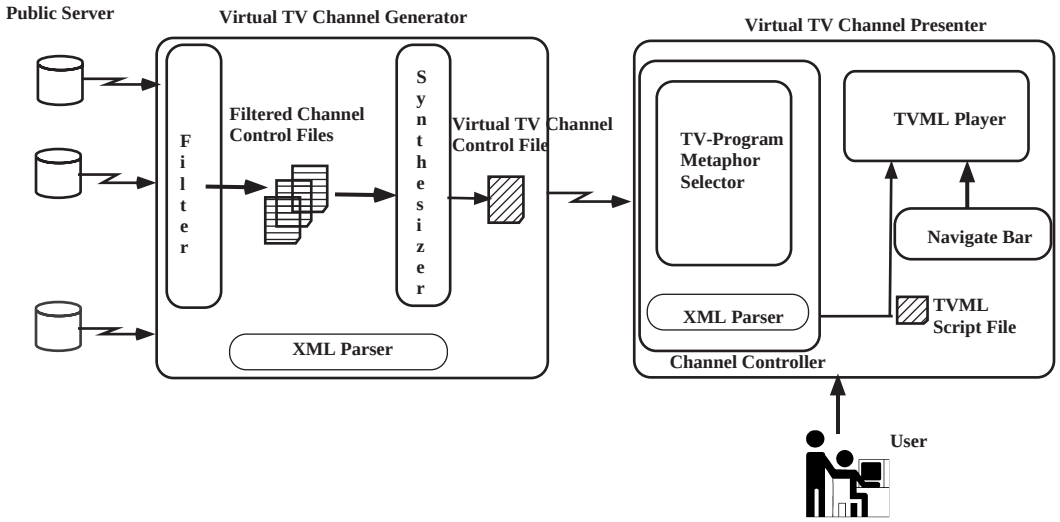

Figure 2.22: System Architecture of Muffin

2.7.4 Prototype System

We developed a prototype system of the virtual TV channel, called *Muffin*. As shown in **Figure** 2.22, *Muffin* has a 3 tier structure:

1. Public Server
2. Virtual TV Channel Generator
3. Virtual TV Channel Presenter

The *public server* means the data sources on the Internet, and stands for a real channel. In our implementation, we assume that the broadcast articles are formatted by XML.

The *Virtual TV Channel Generator* is the core of our prototype system. At first, the *filter* ranks the broadcast information of *public server*, and constructs the higher rank articles as filtered channels, and uses the XML formatted *filtered channel control files* to describe the filtered

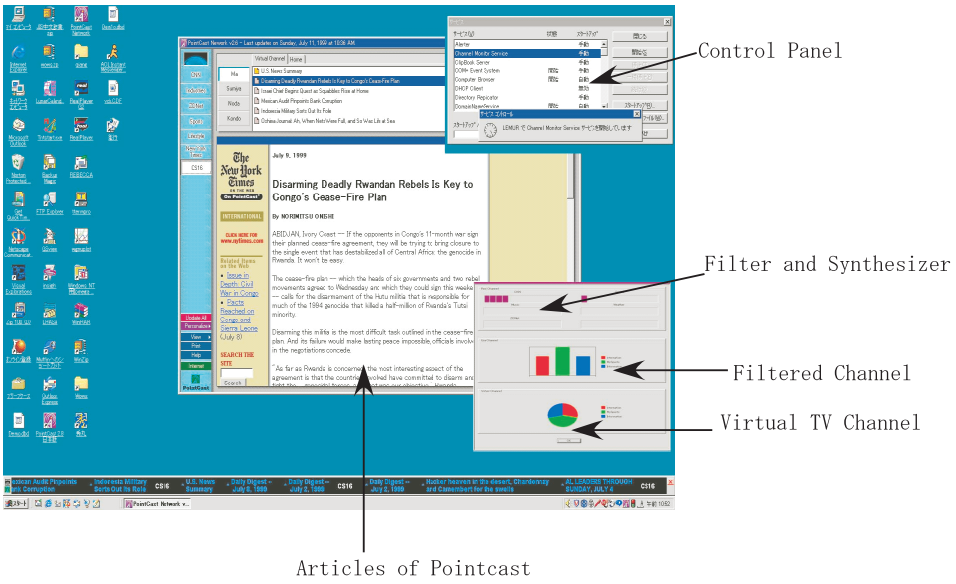

Figure 2.23: Running Example of Virtual Channel Generator

channels. Secondly, the *synthesizer* ranks articles of filtered channel, and selects the higher rank articles into the virtual TV channel control file. The *MSXML*[Microsoft, 1999] was used to parse the source articles, the filtered channel control files and the virtual TV channel control file.

The synthesizer analyzes the virtual TV channel control file as a feedback to itself in order to adjust the ration of filtered channels.

We had developed the virtual TV channel generator on a Windows NT Workstation machine. **Figure** 2.23 is a running example of the current version of virtual TV channel generator.

The virtual TV channel presenter is composed of the *Channel Controller*, *Navigate Bar*, and the *TVML player*. At first, the channel controller receives the virtual TV channel control file from the virtual TV channel generator. Then the *TV-Program metaphor selector* parses the virtual TV channel control file, and creates a scenario: *TVML Script File*. Finally, a TV-program, which represents the articles of the virtual channel, is presented with the TVML player. The navigate bar can be used to control the TVML player just like the remote-control of TV. **Figure** 2.24 is a running example of the current version of presenter.

2.8 Conclusion

Broadcasting-type information dissemination systems on the Internet are becoming increasingly popular due to advances in the area of web technology and information delivery. One of the notable features of push-based, multiple-channel-based information dissemination systems is to deliver information to users in a form of time-series articles. Conventional information retrieval and filtering methods do not consider well the worth of an article from the standpoint of the time-series feature. Moreover, because it is difficult to specify keyword-based query, conventional information retrieval and filtering methods could not discover new valuable information.

We proposed some new criteria (freshness, popularity, and urgency) for query-free informa-

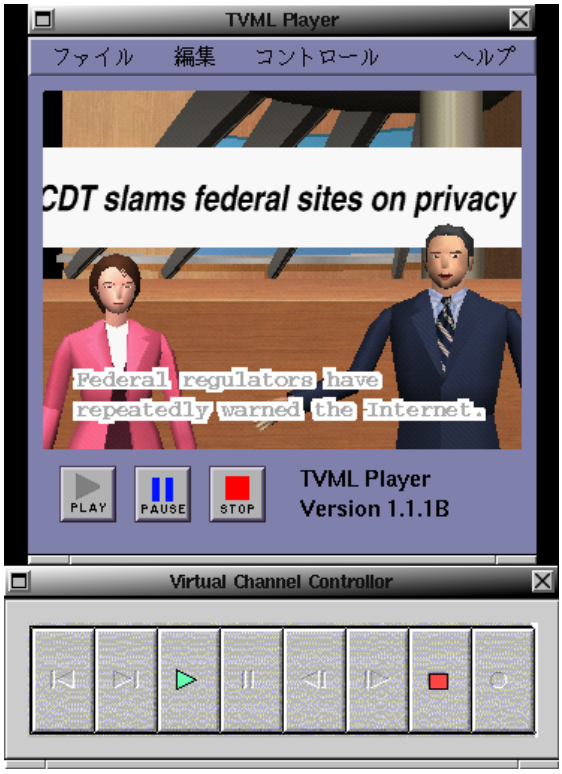

Figure 2.24: Running Example of Virtual Presenter

tion retrieval and filtering. As the evaluation results shown, these criteria and methods are useful for discovering fresh, popular and urgent information from the vast number of incoming articles and web changes.

Based on these criteria, we proposed some application systems. WebSCAN, one of them, is a system that monitors and analyzes the changes of the Web to notify user the important changes by a push-type delivery mechanism. In contrast to earlier works, the important change is picked out based on its change worth estimated by considering both the change content and the structure. Moreover, the notification of important changes is delivered to the users by a push-type mechanism, which separates the user customizing method from the notification to integrate the popularization and personalization.

We also proposed a personal on-line news broadcast system named virtual TV channel for filtering, restructuring and presenting multiple streams of articles. Based on the features of the content of a virtual channel, an appropriate TV-program metaphor (news program, drama, entertainment show, and so on) is automatically selected, and users can enjoy those articles lile a TV-program.

CHAPTER 3

INFORMATION RETRIEVAL BASED ON SPATIAL CRITERION

3.1 Introduction

With the rapidly progressing and spreading of the World Wide Web (WWW), the information about our daily life and residential region is becoming to be more and more active. However, via conventional methods, it is not yet easy to discover such kind of information from the Web. Some portal web sites[Yahoo!regional, 2003; MACHIgoo, 2003] provide directory type search services for regional information. In such portal web sites, the regional information is managed manually. The resource may be limited and some valuable local information may be missed. It's also necessary to define some new retrieval criterions for discovering information about our daily and regional life.

We propose a notion called localness degree for discovering local information from the Web. We say a web page is local when it only interests the residents (users) of some special regions or organizations. From this viewpoint, we compute the localness degree of a web page from two ways: a) estimating its region dependence: the frequency of geographical words and the area of its content coverage, and b) estimating the ubiquitousness of its topic: in other words, we estimate if its information is usual that appeals everywhere and everyday in our daily life.

(a) Region Dependence If a page has high region dependence, it may describe something about a special region and may interest the residents. Therefore, its probability of being local information may be high.

At first, to estimate the region dependence, we compute the frequency of geographical words within a web page. We assume that geographical words are words of region name and organization name. The administration level of each geographical word is considered as a weight value when compute its frequency. If geographical words appear frequently, the localness degree is high.

Secondly, we compute the content coverage of a web page. We plot its geographical words

on a map and compute the area of its content coverage based on MBR (Minimum Bounding Rectangle)[Guttman, 1984; Zaniolo *et al.*, 1997]. If this area is small and there are many geographical words, the localness degree of that web page is considered to be high. In other words, we compute the density of geographical words over the content coverage to estimate the localness degree. If the density is high, localness degree of that page should be high.

Moreover, we compute the document frequency of each geographical word to avoid the effects of usual geographical words on localness degree.

(b) Ubiquitousness of Topic Some events occur everywhere and some events occur only in some special regions. Information about the latter is a kind of scoop and may interest all users (of all regions). In other words, it may be global information more than local information. If a page describes such scoop event (the latter), then it may not be local information. On the other hand, if a page describes ubiquitous event (the former type, high similarity between these events excluding the locations and times), it may only interest users of some special region because that it's usual information. For instance, summer festivals are held everywhere in Japan. These events (summer festivals) are similar although their locations and times may be different. People may be only interested in the summer festival of their living town (or some special places to them, such as his/her hometown). In other words, the summer festival of a special region may interest the residents only and its localness degree could be high.

To estimate the ubiquitousness of a web page, we compare it with web pages describing something about different region to compute their similarities without geographical words and proper nouns. If the number of its similar (excluding geographical words and proper nouns) pages is high, this web page may represent usual information, such as ubiquitous topic or event. Therefore, its localness degree should be high.

Based on the notion of localness degree, we also propose a localness filter for searched web page. The localness filter will compute the localness degree of each web page of search results and select more (or less) local information as the filtering results. For example, if a user wants to get more local information, the localness filter will return the web pages that have high localness degree. On the other hand, if a user wants to exclude local information, the web pages that have lower localness will be returned. In contrast to conventional system, the localness filter is useful to:

- acquire local information, which is not easy to clearly specify in keywords,
- exclude the local information, and
- discover local information over multi-regions.

3.2 Related Work

Mobile Info Search (MIS)[Miura *et al.*, 1998] is a project that proposed a mobile-computing methodology and services for utilizing local information from the Internet. Their system KOKONONET[MIS2, 2002] exchanges the information bi-directionally between the Web and

the real world based on the location information. Our research differs in that we define a new concept, the localness degree of a web page, for discovering 'local' information from the Web automatically.

Buyukkokten et al[Buyukkokten *et al.*, 1999] discussed how to map a web site to a geographical location, and studied the usage of several geographical keys for the purpose of assigning site-level geographical context. By analyzing "whois" records, they built a database that correlates IP addresses and hostnames to approximate physical locations. By combining this information with the hyperlink structure of the Web, they were able to make inferences about the geography of the Web at the granularity of a site.

Geographic Search[Egnor, 2002] adds the ability to search for web pages within a particular geographic locale to traditional keyword searching. To accomplish this, Egnor converted street addresses found within a large corpus of documents to latitude-longitude-based coordinates using the freely available TIGER and FIPS data sources, and built a two-dimensional index of these coordinates. Egnor's system provides an interface that allows the user to augment a keyword search with the ability to restrict matches to within a certain radius of a specified address. In consideration of how much a page has stuck to the area, this point differs from our research. Moreover, our work is focused on how to discover local information from the Web contextually: content and correlation with others.

In the Digital City Kyoto project[Ishida, 2002], KyotoSEARCH[Lee *et al.*, 2002] had been proposed. KyotoSEARCH supports users' information navigations among the Web, the map and web-based geographic knowledge in an integrated way. KyotoSEARCH indexes the web pages by mapping it with its relevant locations on a map and provides a visual interface to help users navigate regional information.

In contrast to these systems and services, the main contribution of our work is that our system can estimate not only which region a web page relates to, but also how 'local' (or how 'ubiquitous') a web page is.

3.3 Localness Degree

As we mentioned before, a web page is local means that it only interests users of some special regions or organizations. We compute the localness of a web page from two ways: 1) region dependence and 2) ubiquitousness of its topic. In other words, if its content bears closely to a special region, the localness of that web page is considered to be high. We also say that a web page is local if its information (topic) is usual that appears everywhere and everyday in our daily life.

3.3.1 Region Dependence

We noticed that a web page with a high dependence on a region includes much geographical information, such as the names of the country, state (prefecture), city, and so on. Therefore, we compute the frequency of these geographical words of a web page to estimate its region dependence.

The content coverage is another factor to estimate a web page's region dependence. If a web

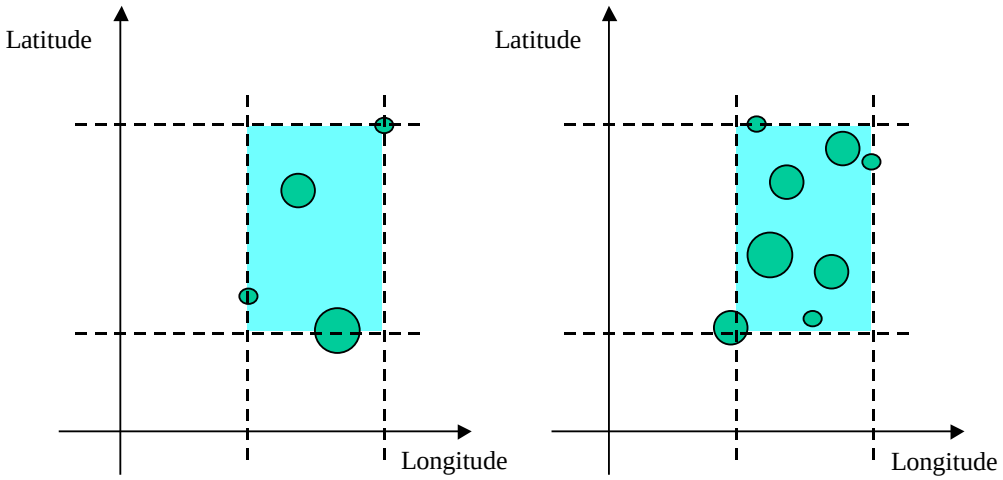

Figure 3.1: Example of Region Dependency Based on MBR

page describes information about a narrow region, its region dependence may be high. In other words, if the content coverage of a web page is very narrow, we say that it is local.

(a) Frequency of Geographical Words A web page has many detailed geographical words; it may deeply depend on some regions. Here, 'detailed geographical word' means that geographical word identifies a detailed location. Usually, we can use the administrative level of each geographical word (region name) as its detailedness [*]. For example, "Kyoto" is more detailed than "Japan". When compute the frequency, we assign weight values to geographical words according to their detailedness. The simplest rule is to set up weight values in the following order: country name $<$ organization name $<$ state (prefecture) $<$ city $<$ town $<$ street (road).

We also compute the document frequency of each geographical word within a web pages' corpus (i.e., web pages in the same web site as the estimating one.) to estimate region dependence of a web page. That's to say, if the document frequency of a geographical word contained in the estimating web page is high, such geographical word is a common one and less affects the localness of that page.

In short, if a web page includes many detailed geographical words that have lower document frequencies, it is local.

(b) Content Coverage We convert the region names and organization names to pairs of location data such as $(latitude, longitude)$ and estimate the content coverage of the web page based on the MBR (Minimum Bounding Rectangle) method[Guttman, 1984; Zaniolo *et al.*, 1997].

Each geographical word is converted to a two-dimensional point (latitude, longitude)[†]. We plot all of these points on a map, on which the y-axis and x-axis are latitude and longitude, respectively. The content coverage of a web page could be approximately computed by the MBR

[*]In our current work, we just consider the detailedness of geographical words based on administrative levels. We also observe that the population density is also important. We will discuss this issue in our future work.

[†]In our current working phase, we use a Japanese location Information database[Takeda, 2000], which includes latitudinal and longitudinal data for the 3,252 cities, towns, and villages in Japan.

(Minimum Bounding Rectangle), which contains these points.

The number and sizes of points plotted on the MBR are also used for estimating the localness. Point size represents the detailedness of a geographical word. A motivation example is illustrated in **Figure** 3.1. Four points are plotted on the left MBR, and eight points are plotted on the right MBR. Even if the areas of the two MBRs are equal, their localness degrees should be different because that the right one may describe more detailed information about that region.

In short, if the content coverage is narrow and there are many detailed geographical words in it, that web page is local. As mentioned before, the document frequency of each geographical word is also important to avoid the effects of common geographical words on localness degree.

In short, we can compute the localness based on region decency as follows:

$$local_d(p) = \frac{\sum_{i=1}^{k} weight(geoword_i) \cdot tf(geoword_i)/df(geoword_i)}{MBR(p) \cdot words(p)} \quad (3.1)$$

$$MBR(p) = (lat_{max} - lat_{min})(long_{max} - long_{min}) \quad (3.2)$$

where $local_{de}(p)$ is the localness degree. $weight(geoword_i)$ is the function used to estimate the detailedness of geographical words. $tf(geoword_i)$ is the frequency of $geoword_i$ within p. $df(geoword_i)$ is the document frequency of $geoword_i$ within a page corpus. MBR_p is the content coverage of page p. $words(p)$ stands for the number of words containing in p. The maximum latitude and longitude are lat_{max} and $long_{max}$, respectively. The minimum latitude and longitude are lat_{min} and $long_{min}$, respectively. When only one point exists in a page and $MBR(p)$ computed by Function (3.2) will be 0. When such exception has been caught, we set $MBR(p)$ to 1. Here, the stand unit of latitude and longitude used to compute $MBR(p)$ is second.

Some detailed place names have no match in the location information database[Takeda, 2000]. In this case, we downgrade the detailedness of such place name to find an approximate location data. For example, if the location data of "C street, B city, A state" is not found, we could use the location data of "B city, A state" as an approximate matching.

Different places may share the same name. To avoid a mismatch, we analyze the page's context to clearly specify its location data. For example, to match up "Futyu city" to its correct location data, we can examine the context. If "Hiroshima" appears around the "Futyu city", we could use the location data of "Futyu city, Hiroshima prefecture" to find the location data of "Futyu city" in this page. On the other hand, if "Tokyo" appears, we should use "Futyu city, Tokyo metropolitan" to get the proper location data.

3.3.2 Ubiquitousness of Topic

Some events occur everywhere and some events occur only in some special regions. Information about the latter is a scoop and may interest all users (of all regions). Oppositely, a ubiquitous occurrence may have a high localness degree. For example, a summer festival (see **Figure** 3.2), an athletic meet, and weekend sale are ubiquitous events that appear usually. A ubiquitous occurrence may be a normal part of our daily life. Therefore, if a page describes scoop event, it may not be local information. On the other hand, if a page describes ubiquitous event, it may be local information.

We estimate the topic ubiquitousness of a web page with two factors: the similarity between pages and the locations where events hold. If the pages are similar in content but different from

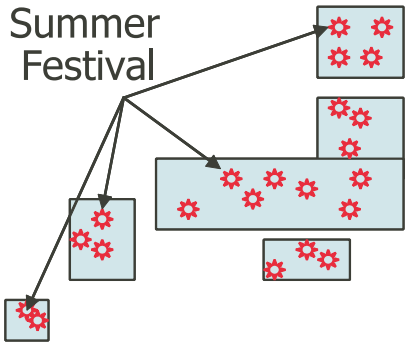

Figure 3.2: Example of Ubiquitous Topic: Summer Festival

location or time, their topics are usual. Here, the similarity between page a and b is calculated as follows based on vector space model[SALTON, 1968]:

$$sim(A,B) = \frac{v(a) \cdot v(b)}{|v(a)||v(b)|} \tag{3.3}$$

where, $v(a)$ and $v(b)$ are keyword (excluding geographical words and proper nouns) vectors of page a and b, respectively.

In short, the localness degree $local_u$ based on the ubiquitousness of topic of web page p is computed as following:

$$local_u(p) = m/n \tag{3.4}$$

where, m is the number of pages, whose similarities (with page p, excluding geographical words and proper nouns) are greater than threshold θ. n is the number of pages, which are used for similarity comparison.

We also noticed that the ubiquitousness of a web page is a relative concept. A web page may be local for a wide region while it's considered to be not local for a very narrow region. For example, although one event rarely occurs within Kyoto city, it may be occur often within the Kansai area. A web page describing such event is usual within Kansai area and it is local. However, it is not usual and not local within Kyoto city. Similarly, although an event which occurs often in summer will be judged to be local within the time scope of summer, it may be not local because it is not usual one within the time scope of one year. Thus, we modify the computation of localness degree based on ubiquitousness. Within time duration τ and region scope Ω, the localness ($local'_u(p,l,\Omega,\tau)$) of web page p based on ubiquitousness at the region level l is computed as follows:

$$local'_u(p,l,\Omega,\tau) = m'/n' = \frac{\sum_{x \in area(\Omega,l)} (countif(x,\theta))}{count(\Omega,l)} \tag{3.5}$$

where, m' is the number of regions containing many similar pages of p at the region level l within region scope Ω and time duration τ. n' is the number of regions at level l within Ω. $area(\Omega,l)$ is the set of regions at level l within Ω. If the ratio of similar pages (of p) among all pages of region

x is bigger than threshold θ, $countif(x,\theta)$ returns 1; else, it returns 0. $count(\Omega,l)$ is the number of regions at region level l within Ω.

3.3.3 Integrated Localness Degree

We also can define an integrated localness degree based on the region dependency and ubiquitousness, such as the sum of $local_d(p)$ and $local'_u(p)$. The integrated localness $local_i(p)$ of page p is computed as follows.

$$local_i(p) = f(local_d(p), local'_u(p)) \tag{3.6}$$

where, f stands for an integration function of $local_d(p)$ and $local'_u(p)$, such as the sum, logical OR, etc.

3.4 Evaluation

3.4.1 Evaluation Environment

To estimate the proposed spatial criteria, we built 3 kinds of filters based on integrated localness degree (filter 3: F3), localness degree based on region dependency (filter 1: F1) and localness degree based on ubiquitousness (filter 2: F2), respectively.

Each filter would select the web pages which had value *true* returned by its filtering function as the local web pages. The details of each filter and their filtering function are described as follows. Hereafter, p stands for web page. $\theta_1, \theta_2, \theta_3$ stand for pre-specified threshold values.

- F1 is a filter based on the region decency based localness degree. The filtering function of F1 is defined as follows.

$$F1(p) = \begin{cases} true & \text{if } local_{f1}(p) \geq \theta_1 \\ false & \text{if } local_{f1}(p) < \theta_1 \end{cases} \tag{3.7}$$

  $local_{f1}$ is computed as follows.

$$local_{f1}(p) = local'_d(p) \tag{3.8}$$

- F2 is a filter based on the localness degree based on ubiquitousness. Its filtering function is defined as follows.

$$F2(p) = \begin{cases} true & \text{if } local_{f2}(p) \geq \theta_2 \\ false & \text{if } local_{f2}(p) < \theta_2 \end{cases} \tag{3.9}$$

  $local_{f2}(p)$ is computed as follows.

$$local_{f2}(p) = local'_u(p,l,\Omega,\tau) \tag{3.10}$$

- F3 is a filter based on one of the integrated localness degree. Here, the integrated localness degree $local_{f3}(p)$ of p is computed as follows.

$$local_{f3}(p) = \alpha \cdot local'_d(p) + \beta \cdot local'_u(p,l,\Omega,\tau) \tag{3.11}$$

Table 3.1: Evaluation Results

Filter	selected pages	relevant pages	selected relevant pages	precision ratio	recall ratio
F1	638	504	357	0.560	0.708
F2	605	525	336	0.555	0.64
F3	780	884	534	0.685	0.604

where, α, β are weight values.

Thus, the filtering function of F3 is defined as follows.

$$F3(p) = \begin{cases} true & \text{if } local_{f3}(p) \geq \theta_3 \\ false & \text{if } local_{f3}(p) < \theta_3 \end{cases} \tag{3.12}$$

We used 1918 news pages collected from ASAHI.COM (http://www.asahi.com/), a well known news web site in Japan, during 3 month period and 47 regions at the region level of prefecture.

Initially, we excluded all of the structural information (e.g., HTML tags) and the advertisement content from the HTML sources of these web pages. We used ChaSen[ChaSen, 2003] for Japanese morphology analysis and only the nouns as keywords for further processing. To exclude stop words, we built a stop words dictionary which contains 593 terms in English and 347 terms in Japanese.

We set the region level to be prefecture. Thus, in this evaluation, $count(\Omega, l) = 47$. The detailedness was defined as a three level value ranging from 1 to 3 as follows: $country < organization\,(prefecture) < city\,(town, village)$. We had used a location database[Takeda, 2000] containing longitude and latitude data of 3252 places in Japan.

Based on some preliminary tests, the parameters used in the filters of this evaluation were $\theta_1 = 0.25, \theta_2 = 0.092, \theta_3 = 0.3$, $\alpha = 1$, and $\beta = 1$. Moreover, we considered two web pages to be similar if their similarity was greater than 0.03 in computing $local_u(p)$.

3.4.2 Evaluation Results

Table 3.1 shows the evaluation results. The numbers of selected pages of F1, F2 and F3 were 638, 605 and 780, respectively. The numbers of relevant pages which had been selected by a user were 504, 525 and 884, respectively. The relevant pages of F1 and F2 were selected by a user from the respective of deeply concerning to a special region or organization and describing usual information, respectively. The relevant pages of F3 were selected as the union of relevant pages of F1 and F2. The recall ratio is the rate of relevant articles selected by each filter among all relevant articles. Precision ratio is the rate of relevant articles selected by each filter among all it selected articles.

The main considerable failure reasons of region dependency computation are described as follows.

- Test data in our evaluation had been pre-organized into regions by ASAHI.COM. These web pages contained a few terms of region names and organization names. We might fail in computing the rate of region names and organization names among all words. For avoiding this kind of failure, using the structure of these web pages is considerable. For example, we could construct these pages into a DAG (directed acyclic graph) and refer the region or organization names appearing in its parent node (web page) when we compute the region dependency of a web page.

- The location database used in our evaluation just contained the longitude and latitude data of main places in Japan, such as seats of municipal governments. Thus, we could not construct well the MBR to compute the content coverage sometimes. Using a database containing more detailed data is necessary.

- We failed in extracting pure body-text of some web pages. In other words, we had gotten many noise terms. For example, a web page may have a frame containing many region names as anchor text. Although such frame is not related to the content of that page, we had extracted these unrelated anchor text as its geographical words. It is necessary to analyze the HTML source and find the pure body text of a web page.

On the other hand, we have failed in compute the ubiquitousness by the following causes.

- We just used web pages during 3-month period and could not find enough similar web pages to compute the ubiquitousness sometimes. Actually, as we mentioned before, ”ubiquitousness” is a relative concept. It is necessary to collect well the comparison collection from time and space (region) perspectives.

- We only tested the news articles. Usually, news articles only report scope and important events, not the usual information. We should use other test data collection to do further evaluation.

- Some web pages contain a small number of words. If we excluded proper nouns from them, we may fail in similarity computation.

From the evaluation results, we could say that the concept of region dependency and ubiquitousness are useful for picking out the concern information to a certain region and the usual information appearing often. In other words, the notion of localness degree is useful to discover information describing our daily and regional life.

3.5 Application System: Localness Filter for Searched Web pages

Although more and more information about our daily life and residential region becomes to be active on the Internet, it’s not yet easy to acquire and exclude such kind of information by conventional information retrieval systems and services.

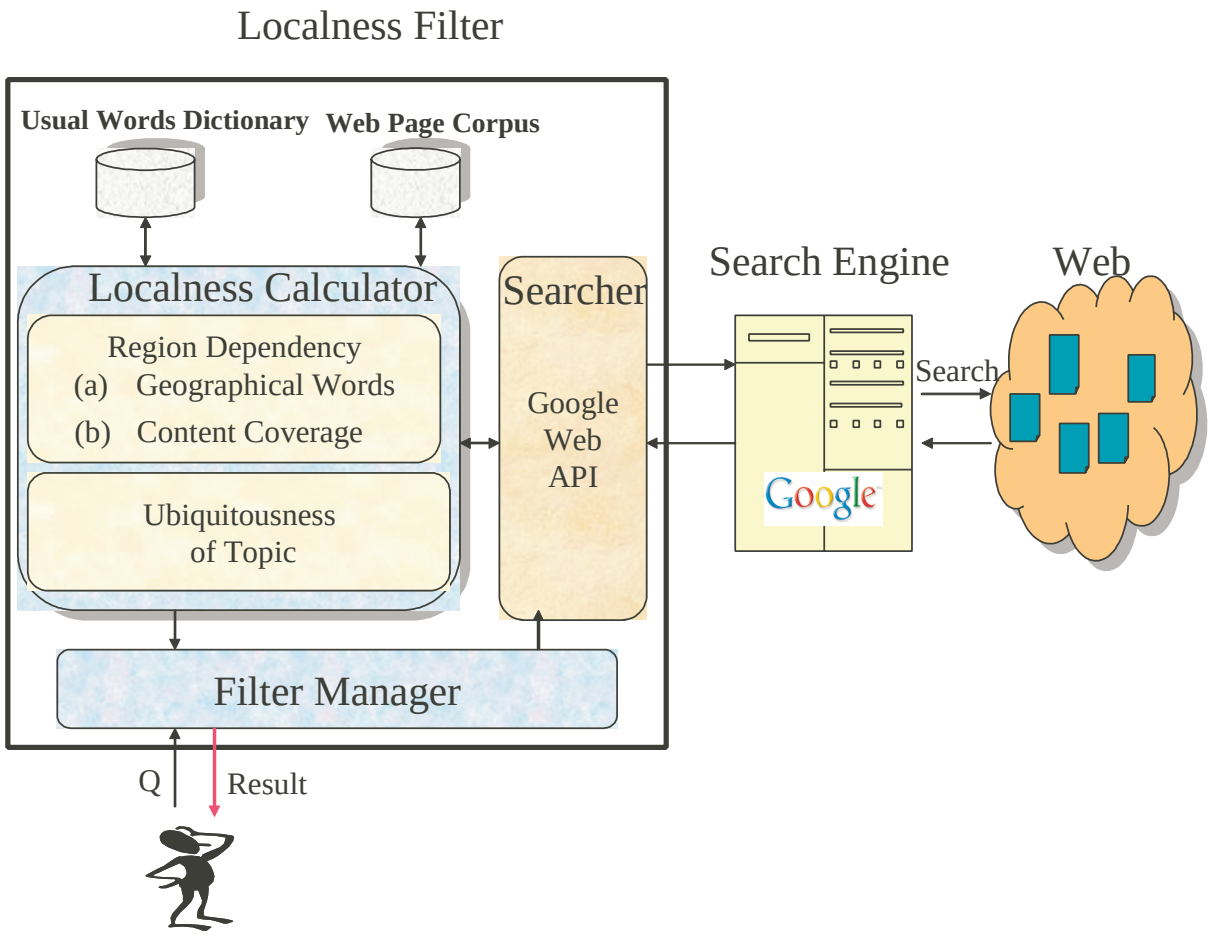

Figure 3.3: Architecture of Localness Filter

In this section, we introduce a localness-filter for searched web pages based on the localness notion. The goal of our localness-filter is to help users to find or exclude local information from the search results returned by search services.

3.5.1 Localness Filter

In the localness filter, the localness degree of page p is computed as follows:

$$local(p) = \alpha \cdot local_d(p) + \beta \cdot local'_u(p) \tag{3.13}$$

where, $0 \leq \alpha \leq 1, 0 \leq \beta \leq 1$ are weight values, which can be specified by a user himself/herself. For example, a user can let $\alpha = 1, \beta = 0$ to select the region dependency based localness $local_d(p)$ as his/her filtering function.

Localness filter has two modes: exclusion mode and inclusion mode. In exclusion mode, the web pages whose localness degrees (computed by function (3.13)) are greater than threshold θ will be excluded from the searched web pages. On the other hand, in the inclusion mode, only the local web pages whose localness degrees are greater than threshold will be the survivors.

The region level of localness is defined as a three-grades value ranging from 1 to 3; 1 means

Figure 3.4: Running Example of Localness Filter

normal level (e.g., according to the level of prefecture.), 2 means high local level (e.g., according to the level of city.) and 3 means very high level (e.g., according to the level of county, ward, village.).

Figure 3.3 illustrates the architecture of localness-filter. The *filter manager* will accept the request of a user and return the results to the user. The *searcher* receives search request from the *filter manager* and accesses the Google Web service[Google, 2003] to get search results. The searched web pages will be passed to *localness calculator* to compute their localness degrees. The parameters for localness computation are set up by *filter manager* based on user's specification. *filter manager* also generates the filtering results according to the filter mode specified by a user. *Web Page Corpus* includes some collections of web pages to compute the document frequencies of geographical words and the topic ubiquitousness of each page.

3.5.2 Prototype System

We developed the prototype system of localness filter on Microsoft .net platform[.net, 2002]. We used ChaSen[ChaSen, 2003] for Japanese morphological analysis to extract geographical words from a web page. **Figure** 3.4 is a running snapshot of our current system. A user can input his/her query into the keywords box and customize the localness filter: filter mode (by ratio buttons), parameters of filtering function and region level (by slider bars). In the results viewer, the results will be sorted in descending order of localness according to the inclusion mode. The results also can be sorted in ascending order according to the exclusion mode. The number of "⋆" signals the degree of localness: high degree of localness will bring up more "⋆". By clicking

the URL in the results viewer, the browser will start up to view the web page.

3.5.3 Evaluation of Localness Filter

We compared the filtering results of localness filter and search results of Google to estimate the localness filter. In the comparison evaluation, we set the parameters of filtering function (3.13) as follows: $\alpha = 1.0, \beta = 1.0$. The level of localness was set to be normal level. The filter mode was set to be inclusion mode. In the comparison evaluation, the evaluation targets were limited to the top 50 pages (ranked by Google) of search results.

The first query for the comparison evaluation was "Noodle"[‡]. By localness filter, 23 pages were accepted as local ones from the top 50 searched pages. The accepted pages were also considered to be local pages by a user. On the other hand, 27 pages were excluded form the searched web pages while 15 pages of them were considered to be local by a user. The recall ratio and precision ratio were 0.605 and 1.0, respectively. The top 10 searched pages' ranking order by Google and localness filter were $(p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10})$ and $(p_4, p_9, p_{10}, p_5, p_2, p_6, p_1, p_7, p_8, p_3)$, respectively. This shows that localness filter can help users to find local information quickly.

Another query for the comparison evaluation was "Chocolate Kobe". On the normal level of localness, 33 pages were accepted as local pages by the localness filter while 11 of them were not local by estimation of a user. On the other hand, 17 pages were excluded while 3 pages of them were selected to be local by a user. The recall ratio and precision ratio were 0.880 and 0.667, respectively. On the high level of localness, 14 pages were accepted including 5 irrelevant (estimated by a user) pages. On the very high level of localness, 7 pages were accepted including 1 irrelevant page. The ranking order of top 10 searched pages by localness filter was $(p_8, p_6, p_5, p_7, p_4, p_3, p_{10}, p_9, p_1, p_2)$.

Since our evaluation was limited, there are more works needed to be done. Nevertheless, from the comparison evaluation, at least, we could assume that localness is useful for filtering search results to discover local information. We also could assume that the level feature of localness is helpful for filtering search results from the query with some keywords. These will be tested in our future evaluations.

3.6 Conclusion

There are many web pages, which provide information about our daily life and residential region. These pages are very important for residents and may only interest them. In this chapter, we proposed a query-free information retrieval method based on the notion of *localness*. With this method, users can acquire or exclude more local information from the Web according to their intentions. We estimate a web page's localness from two aspects: region dependence and topic ubiquitousness.

We also noticed that users' location information is useful to estimate the localness of a web page. If a web page was often accessed by users from the same region, and there was little access from other regions, we could say that such page only interests users of that special region and

[‡]In our comparison evaluation, the keywords of query were in Japanese.

its localness is high. On the other hand, if some information was published only by the users from a same region, and there were few users reporting the same information in other regions, we could say that such information is local because that it only interests users of a special region. We also can compute the frequency of daily words within a web page to estimate its ubiquitousness by referring a pre-specified usual (daily) words dictionary. If the frequency is high, we say that web page describes usual information and its ubiquitousness is high. The usual words dictionary could be constructed from a corpus of user-selected web pages that describe usual information. We will discuss these issues in our future work.

Further evaluations of the localness notion and the localness filter need to be carried out, such as comparison evaluations with dictionary-type search in portal region web sites (Yahoo! Region[Yahoo!regional, 2003], MACHIGoo[MACHIgoo, 2003], etc.) and keywords search in agent-type search services (Google, etc.). Further refinements on the localness notion and localness filter are also planned to do as our future work, such as, considering well the hierarchical relationships among geographical words to estimate the region dependence and so on.

CHAPTER 4

COMPLEMENTARY INFORMATION RETRIEVAL FOR INFORMATION AUGMENTATION

4.1 Introduction

The integration of information derived from different types of media is becoming increasingly more important and useful in acquiring detailed information. For instance, although a TV program may be of excellent quality and extremely realistic, its time restrictions (on-air time) and obligation to concede to popular opinion (serving the public interest) limit the details and horizons of its information. However, information published via the Internet is diverse and has few restrictions. For example, there may be many different articles (web pages) from diverse perspectives, although they report the same event. Yet neither TV programs nor web pages can solely satisfy our needs at times. The intuitive integration of the TV and the Internet, such as synchronizing a TV program with its related web pages, could improve the quality of information and broaden our horizons because each complements the other. To find the complementary information of a given web page or video, in this thesis, we propose a novel complementary information retrieval mechanism for information augmentation based on a notion called topic structure.

We call an event or an activity a ”topic”. In order to represent such ”topics” described in a video or a web page, we use a notion of topic structure. Intuitively, a topic structure is made up of a pair of subject and content terms. The subject terms denote the most dominate terms of a web page or a text stream (keyword sequence, e.g., closed captions of videos.). A content term means a term which has strong co-occurrence relationships with the subject terms. Naturally, the subject and content terms should appear in the web page (or the text stream). Based on the topic structure notion, we propose a content join operation for information augmentation. In this thesis, the topic structure is represented as a connected DAG (Directed Acyclic Graph) and the join between two topic structures is represented as a union of their graphs.

We extract the topic structure of a given web page or video and then find other web pages or videos which provide complementary information on the given one. Based on the content join, we generate some queries to find the candidate web pages or videos which may provide additional information on a given web page or video. In order to select the most proper one from the search results, we also use a notion called complementarity degree defined based on computing the difference between the join results and the given example to rank the search results.

Our complementary information retrieval mechanism is similar to the query-by-example[Zloof, 1977] technology, but differs in that not just finding the similar web pages but complementary web pages which provide additional information on the given example. It is to say, the searched web page by our mechanism can detail the given example or describe it from different perspectives.

4.2 Related Work

Query by example (QBE)[Zloof, 1977] is a method of query creation that allows the user to search for similar documents based on an example in the form of a selected text string or in the form of a document name or a list of documents. The complementary information retrieval mechanism is similar to the conventional QBE systems because it also formulates the query based on a given web page or video. However, in contrast to the conventional systems, we generate the queries based on the topic structure of the given example to find its complementary web pages. These searched web pages are not just similar to the given example but also provide additional information.

[Henzinger *et al.*, 2003] proposed some methods to automatically generate query from the closed captions to find similar pages of TV-programs from the Web. In contrast to [Henzinger *et al.*, 2003], in our mechanism, the searched web pages are not just similar to the given example (TV-program). They can detail its content or describe it from different perspectives.

TopicMap [TopicMap.org, 2003] is a new ISO standard for organizing, retrieving, and navigating information resources. It provides powerful new ways of navigating large and interconnected corpora. Here, names, resources, and relationships are said to be characteristics of abstract subjects that are called topics. In contrast to TopicMap, the notion of topic structure, which is proposed in this thesis, is used to describe the subject and content of an information resource through keywords. We also extract the topic structure automatically instead of manually through user specifications. TDT (Topic Detection and Tracking)[Wayne, 2000] research develops algorithms for discovering and threading together topically related material in streams of data such as newswire and broadcast news. In TDT, the notion of ”topic” is defined as a seminal event or activity, along with all directly related events and activities. Therefore, a topic is structured as multi-sequential stories (events and activities, etc.) in TDT. Its ”story” notion is similar to our ”topic” notion in this thesis. However, in our work, each event (or activity) is considered as a separate ”topic” although it may be related to the others. We represent the content of a ”topic” by using a topic structure, which is a pair of subject and content terms.

Join is a fundamental query operation in the relational database for information retrieval obtained from two different relations based on the Cartesian product of these two relations[Mishra

and Eich, 1992]. Guha et al. [Guha *et al.*, 2002] proposed an approximate match in structure and content for integrating XML data sources. In this thesis, content join operation based on the topic structure notion, are used to formalize the information augmentation process. Moreover, based on the proposed join operation, we search for information which can detail the original information and broaden its coverage.

Shasha et al. introduced some algorithms and applications for tree and graph searching and they studied on how to match (or approximately match) a query tree (query graph) to XML data[Shasha *et al.*, 2002] . In contrast, we propose the concept of a topic structure and represent it as a connected DAG through which we propose a join model for integration of web-page and data-stream information.

Data stream processing has been studied for a number of years now and recently. It has attracted renewed interest within the database community[Babcock *et al.*, 2002; Carney *et al.*, 2002]. Because data is appended continuously, it is difficult to handle the whole body of data. Hence, segmentation is a very important issue in data stream processing, especially for block operations (e.g.,*sort* and *max*). Sliding window functions [Sullivan and Heybey, 1998] are very useful in dividing the data stream into sequential sub-streams based on time intervals or data size. [Tucker *et al.*, 2002] proposed punctuation to mark the end of sub-streams in a stream. Here, we propose a content-based approach to automatically segment text streams (especially, closed captions of TV programs or videos).

The issues with video segmentation based on signal recognition have been extensively investigated and most methods are very time consuming[Maybury, 1997]. The Infomedia[Wactlar, 2000] and Mani[Mani *et al.*, 1997] introduced various methods of video segmentation based on analysis of closed captions. However, because these need the whole body of data to be scanned, they are not effective for data streams, which are received continuously.

Some approaches which extract the relationship between keywords appearing in the title and body for information integration have been introduced[Maeda *et al.*, 1997; Murakami and Hirata, 2001]. They focused on the issue that how to organize information in same type media. Like our work, they also considered well the relationship of keywords appearing in title and body. In contrast, we propose a notion of topic structure to represent the different roles of such kinds of keywords. Moreover, we propose a complementary information retrieval mechanism for information integration of different media, such as the TV and the Web.

4.3 Topic Structure

4.3.1 Topic Structure

As previously mentioned, we have considered a topic structure to be a pair of subject and content terms. To reiterate, the subject terms denote the most dominate terms of a web page or video, while the content terms are those terms which have strong co-occurrence relationships with the subject terms. In other words, the subject terms are the centric keywords that play the title role on a web page (or video). The content terms play the body-role on a web page (or video). Both subject term and content term shall appear in the web page or video. A primitive

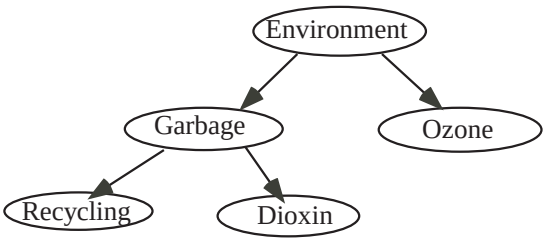

Figure 4.1: Example of Topic Graph

topic structure definition in [Matsukura *et al.*, 2001] is defined as follows:

$$topic = (S, C) \tag{4.1}$$

where, S and C are sets of subject-term* and content-term, respectively.

As an extension, we have assumed that topic structure is hierarchical, i.e., one topic may consist of several sub-topics. Each sub-topic describes some portion of the main topic. In other words, S (named subject element set) or C (named content element set) can contain some topic structures as its member. Our extended topic structure notion is defined as follows.

$$\begin{aligned} topic &:= (S, C) \\ S &:= (subject\text{-}term | topic)^{+} \\ C &:= (content\text{-}term | topic)^{+} \\ subject\text{-}term &:= keyword \\ content\text{-}term &:= keyword \end{aligned} \tag{4.2}$$

Where, "|" stands for "or" and "+" means that the element(s) should appear more than one time. In addition, a keyword should not occur more than one time in a topic structure.

4.3.2 Topic Graph

A topic structure can be represented as a connected DAG (Directed Acyclic Graph) which has at least two vertices: one stands for subject-term and one stands for content-term. Hereafter, we call such graph a topic graph. In a topic graph, a vertex represents a keyword. A directed-edge represents the subject-content relationship of two keywords: the source vertex denotes the subject-term and the destination vertex denotes the content-term.

Definition 1 (Topic Graph) *Given a topic structure t, its topic graph $G(t)$ is defined as follows:*

$$G(t) = (V, E) \tag{4.3}$$

where, V is a vertex set which represents the keywords within t. $E(\subseteq V \times V)$ is a directed-edge set. A directed edge $e = (u, v)$ represents the subject-content relationship between keyword u and v. $\|V\| \geq 2, E \neq \emptyset$.

*[Matsukura *et al.*, 2001] call it "thematic term".

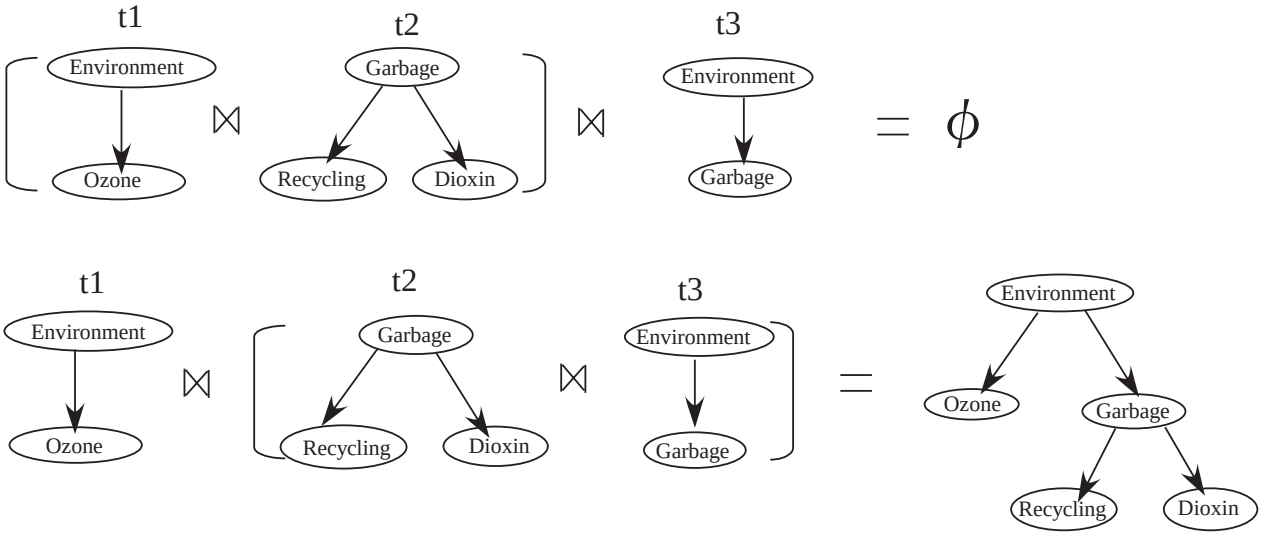

Figure 4.2: Example of Join

Figure 4.1 shows an example of topic graph. In this example,

$$V = \{Environment, Garbage, Ozone, Recycling, Dioxin\},$$
$$E = \{(Environment, Garbage), (Environment, Ozone), (Garbage, Recycling), (Garbage, Dioxin)\}.$$

This topic graph represents the topic structure

$$t = (\{Environment\}, \{(\{Garbage\}, \{Recycling, Dioxin\}), Ozone\}).$$

4.3.3 Topic-structure-based Join

The integration of information derived from multimedia can be regarded as a join, which combines related content from different media such as the TV and the Web. This perspective, based on the topic structure, allows us to define the join of two topic structures to formalize the information integration.

Definition 2 (Topic-structure-based Join) *The join of two topic structures, t and t', means the union of their topic graphs such that result is a connected DAG.*

$$t \bowtie t' = \begin{cases} G(t) \cup G(t'), & \text{if } G(t) \cup G(t') \text{ is a connected DAG.} \\ \phi, & \text{others} \end{cases} \tag{4.4}$$

where $G(t)$ and $G(t')$ stand for the respective topic graphs of t and t'. ϕ stands for empty.

In this definition, we have restricted the join result to a connected DAG. That is to say, the join result of two topic structures should be a single topic structure too. Additionally, $t \bowtie \phi = \phi$. It is obvious that topic structure based join is commutative, i.e., $t_1 \bowtie t_2 = t_2 \bowtie t_1$. Topic structure based join is not associative, for example, as shown in **Figure** 4.2, $(t_1 \bowtie t_2) \bowtie t_3 \neq t_1 \bowtie (t_2 \bowtie t_3)$.

If the join of two topic structures is not ϕ, we say that two topic structures are joinable. Two web pages which have some joinable topic structures are complementary to each other.

4.3.4 Properties of Topic-structure Based Join

In this section, we formalize the complementary information retrieval mechanism based on the topic-structure join model and discuss its properties.

Definition 3 (Join of Topic Structure Sets) *Given two sets of topic-structures, T and T', the join of T and T' is defined as follows:*

$$T \bowtie T' = \{x \bowtie y | x \in T, y \in T'\} \tag{4.5}$$

A web page may consist of several topics; that is to say, the topics of such web page should be represented as a set of topic structures. For example, web page p with two topic structures, t_1 and t_2, could be represented as $t_p = \{t_1, t_2\}$. Therefore, web pages are a set of set of topic structure. In this thesis, to explain simply, we use a set of topic structures to represent the topics of web pages.

$$T_P = t_{p_1} \cup t_{p_2} \cup ... \cup t_{p_n} \tag{4.6}$$

where T_P stands for the topic structures of web page set P. t_{p_i} stands for topic structures of web page p_i, which is a member of P.

We also could represent the topics of a text stream as a set of topic structure, too.

$$T_S = \{t_{s_1}, t_{s_2}, ..., t_{s_n}\} \tag{4.7}$$

where T_S stands for the topic structures of the text stream $S(= s_0 s_1 ... s_n)$. t_{s_i} stands for the topic structure of sub-stream s_i.

Therefore, the integration of text streams and web pages can be formalized as the join of two topic structure sets defined above.

The join of topic structure sets has the following properties. Hereafter, T, T', S, S' stand for topic structure set, respectively. x, y stand for topic structure, respectively.

1. **commutative:** $T \bowtie S = S \bowtie T$. It is to say, the join of topic structure sets satisfies the commutative law. By the definition, it is trivial.

2. **distributive:** $(T \cup T') \bowtie S = (T \bowtie S) \cup (T' \bowtie S)$.
   Proof.

$$\begin{aligned} (T \cup T') \bowtie S &= \{x \bowtie y | x \in T \cup T', y \in S\} \\ &= \{x \bowtie y | x \in T, y \in S\} \cup \{x \bowtie y | x \in T', y \in S\} \\ &= (T \bowtie S) \cup (T' \bowtie S) \quad \blacksquare \end{aligned} \tag{4.8}$$

   Obviously, $(T \cup T') \bowtie (S \cup S') = (T \bowtie S) \cup (T \bowtie S') \cup (T' \bowtie S) \cup (T' \bowtie S')$. In other words, the join of topic structure sets satisfies distributive law.

3. **increasing:** $T \bowtie S \subseteq T \bowtie (T \bowtie S)$. .
   Proof.

(a) if $x \bowtie y = \phi$ then $x \bowtie (x \bowtie y) = \phi$. Hence, $x \bowtie (x \bowtie y) = x \bowtie y$

(b) If $x \bowtie y \neq \phi$ then $x \bowtie (x \bowtie y) \neq \phi$, hence, $x \bowtie (x \bowtie y) = G(x) \cup (G(x) \cup G(y)) = G(x) \cup G(y) = x \bowtie y$

By (a) and (b), $x \bowtie (x \bowtie y) = x \bowtie y$. Hence,

$$\begin{aligned} T \bowtie (T \bowtie S) &= \{x \bowtie (x \bowtie y) | x \in T, y \in S\} \cup \{x \bowtie (x' \bowtie y) | x \in T, x' \in T, y \in S, x \neq x'\} \\ &= \{x \bowtie y | x \in T, y \in S\} \cup \{x \bowtie (x' \bowtie y) | x \in T, x' \in T, y \in S, x \neq x'\} \\ &\supseteq T \bowtie S \quad \blacksquare \end{aligned} \tag{4.9}$$

This property means, the join of topic structure sets does not satisfy the absorption law.

4. **sequential increasing:** $(T \cup T') \bowtie S \subseteq ((T \bowtie S) \cup T') \bowtie S$.
Proof.

$$\begin{aligned} ((T \bowtie S) \cup T') \bowtie S &= ((T \bowtie S) \bowtie S) \cup (T' \bowtie S) && (4.10) \\ &\supseteq (T \bowtie S) \cup (T' \bowtie S) \\ &= (T \cup T') \bowtie S \quad \blacksquare && (4.11) \end{aligned}$$

For the join of topic structure sets T and S, we can divide the topic structure set S (or T) into several arbitrary sub-sets and join each sub-set with the other set T (or S) before the results are merged. By the *distributive* property, it has the same results as that we join the two sets (T and S) in bulk. In other words, the batch processing of topic structure sets join $(T \cup T') \bowtie S)$ and the distributed processing $((T \bowtie S) \cup (T \bowtie S'))$ are equivalent.

The *sequential increasing* property means, sequential processing $((T \bowtie S) \cup T') \bowtie S)$, in which the previous joining results $(T \bowtie S)$ and the new topic structure set (T') are merged into one set before joining with S, is not equivalent to batch processing $((T \cup T') \bowtie S)$. It also means sequential processing $((T \bowtie S) \cup T') \bowtie S)$ is not equivalent to distributed processing $((T \bowtie S) \cup (T' \bowtie S))$. The *sequential increasing* property signifies that the results of sequential processing are larger than and include the results of batch processing and distributed processing.

4.3.5 Complementarity Degree

Basically, the complementarity degree can be computed by a comparison between the topic graphs of the given one and join result.

Definition 4 (Complementarity Degree) *The complementarity degree of the topic structure t' to a given topic structure t is, the sum of differences of the height and width between topic graphs of the join result $(t \bowtie t')$ and t.*[†]

[†]Intuitively, we also can compute the complementarity degree as the product of the width difference and height difference.

$$\begin{aligned} comple(t,t') &= \alpha * (Height(G(t \bowtie t')) - Height(G(t))) \\ &+ \beta * (Width(G(t \bowtie t')) - Width(G(t))) \end{aligned} \tag{4.12}$$

where, α and β are weight parameters. $Height(G(x))$ and $Width(G(x))$ mean the height and width of the topic graph of topic structure x, respectively.

Actually, the difference of width is used to compute the complementarity degree from the perspective of broadening information coverage. On the other hand, the difference of height is used to compute the complementarity degree from the perspective of detailing the given example. Therefore, the parameters α and β stand for the unit values of difference of height and width from the perspectives of detailing and broadening information coverage, respectively.

4.4 Topic-structure Extraction

4.4.1 Co-occurrence Relationship

In this thesis, to extract the topic structure from a given example, we defined two kinds of co-occurrence relationships: 1) undirected term co-occurrence rate and 2) directed term co-occurrence rate.

Definition 5 (Undirected Term Co-occurrence Rate) *The rate of undirected term co-occurrence is a notion used to estimate the co-occurrence relationship between two words. When words w_1 and w_2 co-occur frequently within a topic corpus, we can say that the two words have a strong co-occurrence relationship and that their co-occurrence rate is high. In this thesis, we estimate the undirected term co-occurrence rate $cooc(w_i, w_j)$ between words w_i and w_j using the following function.*

$$cooc(w_i, w_j) = \frac{df(\{w_i, w_j\})}{df(\{w_i\}) + df(\{w_j\}) - df(\{w_i, w_j\})} \tag{4.13}$$

where $df(\{w_i\})$ is the number of topics containing word w_i within a pre-specified topic corpus, and $df(\{w_i, w_j\})$ is the number of topics containing both w_i and w_j.

Definition 6 (Directed Term Co-occurrence Rate) *Within a topic corpus, the directed term co-occurrence rate $\overrightarrow{cooc}(w_i w_j)$ is defined as the rate of occurrence of topics containing keyword w_j in topics containing keyword w_i. We formulated this definition as follows:*

$$\overrightarrow{cooc}(w_i w_j) = df(\{w_i, w_j\}) / df(\{w_i\}) \tag{4.14}$$

where $df(\{w_i, w_j\})$ is the number of topics containing both w_i and w_j, and $df(\{w_i\})$ is the number of topics containing w_i.

Although $cooc(w_i, w_j) = cooc(w_j, w_i)$, $\overrightarrow{cooc}(w_i w_j)$ and $\overrightarrow{cooc}(w_j w_i)$ differ more often than not.

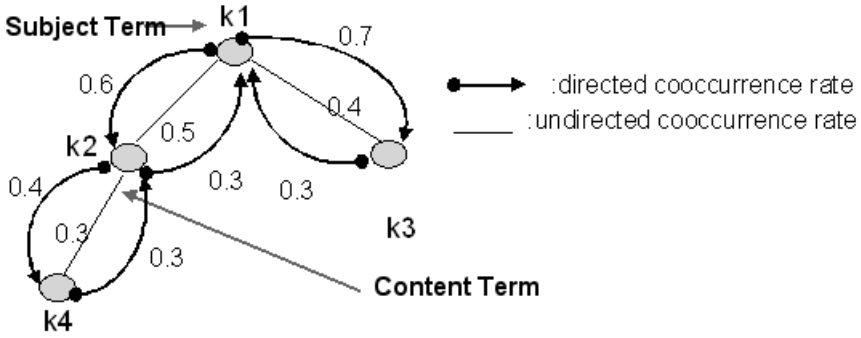

Figure 4.3: Example of Extraction of Subject and Content Terms (not considering the term frequency)

4.4.2 Subject and Content Degrees

We define a notion of "subject degree" to determine whether a keyword had a high probability of being a subject term. The subject degree of word w_i is defined by: 1) its directed term co-occurrence rates with other keywords, and 2) its term frequency. That is, if a keyword has high rates of directed co-occurrence with other keywords within a topic and its term frequency is higher than that of the other keywords, it is considered to be the subject term. We also define a notion of "content degree" to determine the content-terms of a topic based on the undirected co-occurrence relationship.

Definition 7 (Subject Degree) *The subject degree $sub(w_i)$ of keyword w_i within a topic is defined as follows:*

$$sub(w_i) = tf(w_i) + \sum_{j=1, j\neq i}^{n} \overrightarrow{cooc}(w_i w_j) \tag{4.15}$$

where $tf(w_i)$ is the term frequency of words w_i, $\overrightarrow{cooc}(w_i w_j)$ is the directed term co-occurrence rate of words w_i and w_j, and n is the number of keywords within that topic.

Definition 8 (Content Degree) *The content degree $con(w_i)$ of keyword w_i within a topic is defined as the sum of undirected term co-occurrence rates with the subject terms of w_i.*

$$con(w_i) = \sum_{w_j \in S} cooc(w_i, w_j) \tag{4.16}$$

where, S is the subject-term set.

Based on the notion of subject degree, we can determine subject and content terms for a topic. The keywords, which have higher subject degrees (top M, etc.), will be considered to be the subject terms. Consequently, we can extract content terms based on the content degree. That is, when we rank (in descending order) the keywords of the topic (excluding the subject terms) by their content degrees, the top N keywords will be chosen as the content terms. In the example

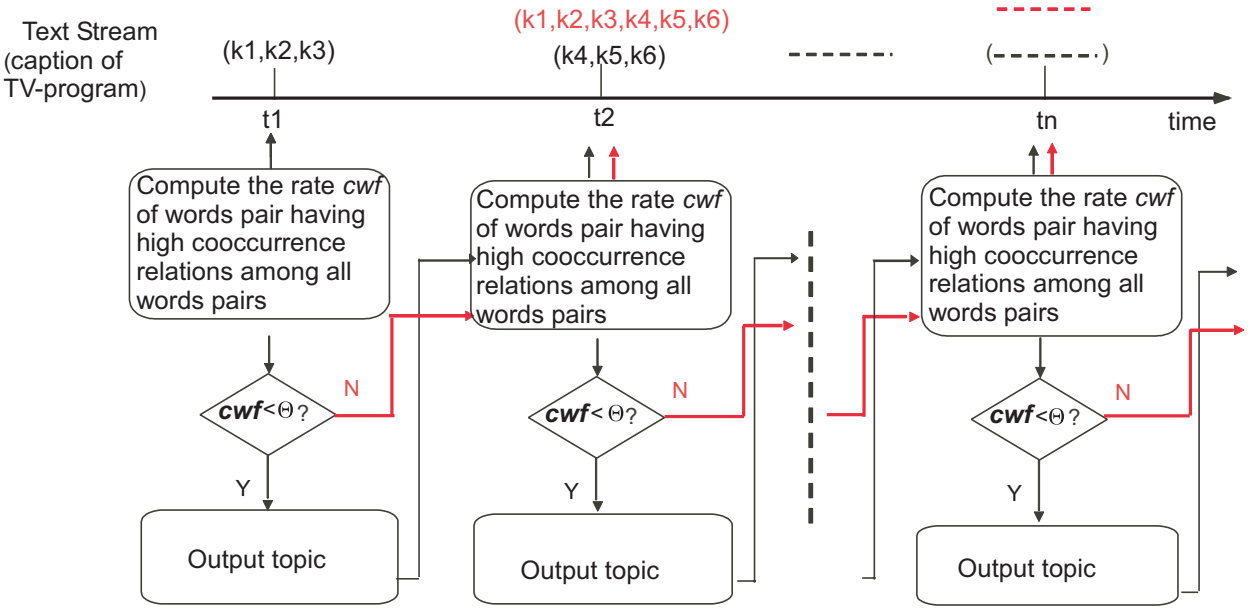

Figure 4.4: Online Topic Detection from Text Streams (t_i is the time point when we received closed captions)

shown in **Figure** 4.3, k_2 has the highest undirected term co-occurrence rates (0.5) with the subject term $k1$. Hence, $k2$ will be selected as the content term (in this example, $M = N = 1$).

If a given web page or video (with closed captions) just contains one topic, we can use the notions of subject degree and content degree to extract its topic structure without further work. In this thesis, we assume that each given example (web page, video, etc) used for query generation just has one topic. If a user given an example having multiple topics, we can just generate queries for each individual topic or require the user to specify his/her interesting topic explicitly.

4.4.3 Topic-structure Extraction from Web Page

As we mentioned before, a web page may contain more than one topic. In other words, a web page can be represented as a topic structure set. To extract the topic structure set of a web page, at first, we assume each paragraph of a web page has only one topic and extract its topic structure. Then we merge the joinable topic structures within these topic structures by using the reduction function defined as follows to construct the topic structure set of a web page.

Definition 9 (Reduction of Topic Structure Set) *Given a topic structure set $T = \{t_1, t_2, ..., t_n\}$, its reduction, $R(T)$, means a graph union of its members.*

$$R(T) = G(t_1) \cup G(t_2) \cup ... \cup G(t_n) \tag{4.17}$$

where $G(t_i)$ is the topic graph of topic structure t_i. $G(t_1) \cup G(t_2) \cup ... \cup G(t_n)$ is a DAG.

4.4.4 Topic-structure Extraction from Text Stream

Since a text stream (closed captions of a video) may consist of multiple topics, at first, we need to detect topics from it. **Figure** 4.4 illustrates the flow of topic detection procedure. Here, we use

the closed captions as the text stream. The basic idea is that if the rate of keyword-pairs with high undirected co-occurrence rates (pre-computed in topic corpus) among all keyword-pairs within various closed captions is high, then these captions belong to one topic.

The detailed procedure is described as follows. Here, CT_i is the keyword set used to detect a topic at time point t_i. ST and ET are the initial and terminal time points of a segmented topic, respectively.

1. Let $CT_0 = \emptyset, ST = 0$.
2. Receive closed captions. If there are no other closed captions, stop.
3. After receiving closed captions at time point t_i ($i \geq 1$), extract keyword set K from the closed captions.
4. Let $CT_i = CT_{i-1} \cup K$.
5. Compute rate $cwf(t_i)$ of keyword-pairs with high co-occurrence rates among all keyword-pairs within CT_i. Here, a high co-occurrence rate means that the undirected co-occurrence rate of the two keywords is greater than a pre-specified threshold θ. m is the number of keywords included in CT_i.

$$cwf(t_i) = \sum_{j=1,k=j+1}^{j=m-1,k=m} cr(w_j, w_k) / \frac{m \cdot (m-1)}{2} \tag{4.18}$$

$$cr(w_j, w_k) = \begin{cases} 1 & \text{if } cooc(w_j, w_k) \geq \theta \\ 0 & \text{if } cooc(w_j, w_k) < \theta \end{cases} \tag{4.19}$$

6. If $cwf(t_i) > \Theta$, go to 9. Θ is a pre-specified threshold.
7. Let $ET = t_i$. Output topic $topic_i$, whose initial and terminal time points are ST and ET, respectively. The keywords of such topic are also outputted for further processing.
8. Let $CT_i = \emptyset, ST = t_i$.
9. Receive closed captions. If there are no other closed captions and $CT_i = \emptyset$, then stop. If there are no other closed captions but $CT_i \neq \emptyset$ then go to 7. Otherwise, go to 3.

Intuitionally, we also can compute the co-occurrence relationships of keywords belonging to adjoining closed captions. If such co-occurrence relationships are weak, we could say that the adjoining closed captions describe different topics. However, such method needs to pre-receive the next closed caption. Moreover, it requires that each closed caption should be one semantic unit, or at least one whole sentence. There are many cases where such requirements cannot be satisfied, such as that we can only receive some portions of a sentence sometimes.

When a topic is detected, we can extract its topic structure based on the subject-degree and content-degree. Moreover, if necessary, we could use the reduction function to reduce the topic structures of a text stream.

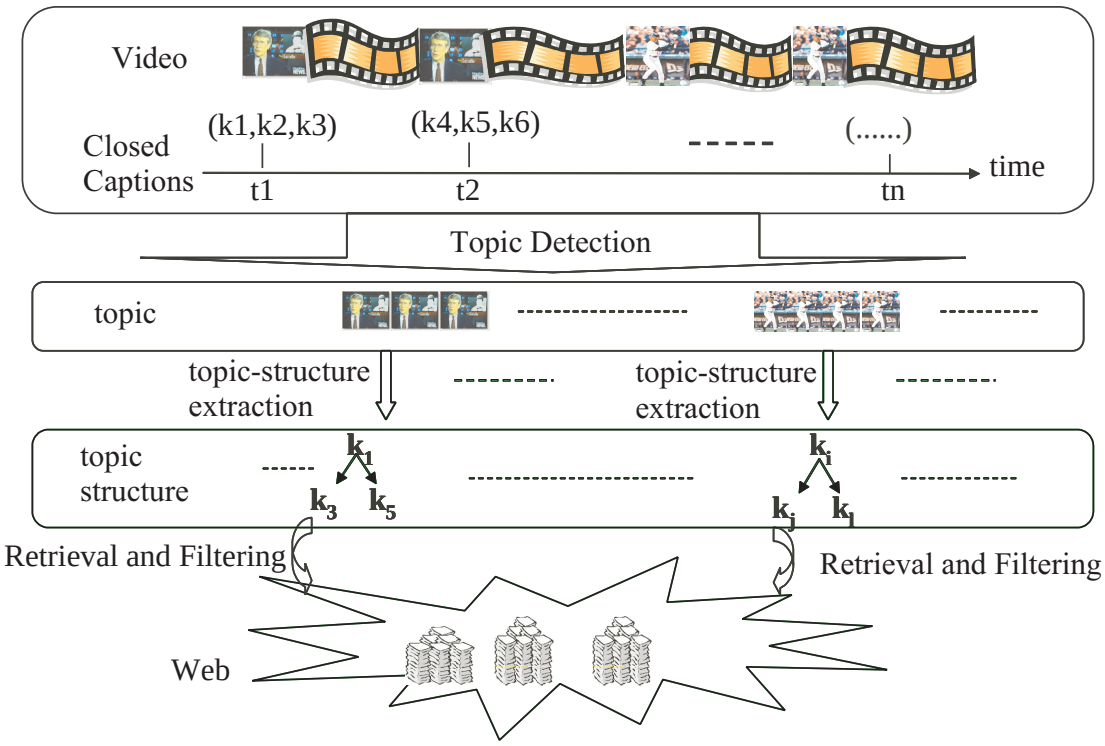

Figure 4.5: Concept of Complementary Information Retrieval Method

4.5 Complementary Information Retrieval

The complementary information retrieval consists of three phases: 1) topic structure extraction from a given example (web page, video, etc.), 2) query generation by using the extracted topic structure, and 3) web retrieval and ranking the searched web pages. **Figure** 4.5 illustrates the concept of complementary information retrieval method.

4.5.1 Query Generation

Here, we assume that the subject element set and content element set of the given topic structure do not have other topic structure as a member. The generated query is used to find the web pages that contain a topic structure which is joinable with that of the given example. Because the join result of two joinable topic structures provide more detailed information than the original ones, we say that each topic structure is complementary to the other.

[Oyama and Tanaka, 2003] reported that it is useful to extract the topic structures of a web page by using the ”title” and ”body” tags. Based on this work, we roughly consider that the keywords appearing in title and body of a web page are its subject and content terms, respectively.

We defined four kinds of queries to find the related web pages of a given example: 1) CD (content-deepening) query, 2) SD (subject-deepening) query, 3) SB (subject-broadening) query, and 4) CB (content-broadening) query.

CD query and SD query are based on the join such that the subject terms in one topic structure appear as the content terms in the other. Such join will add details to the original information.

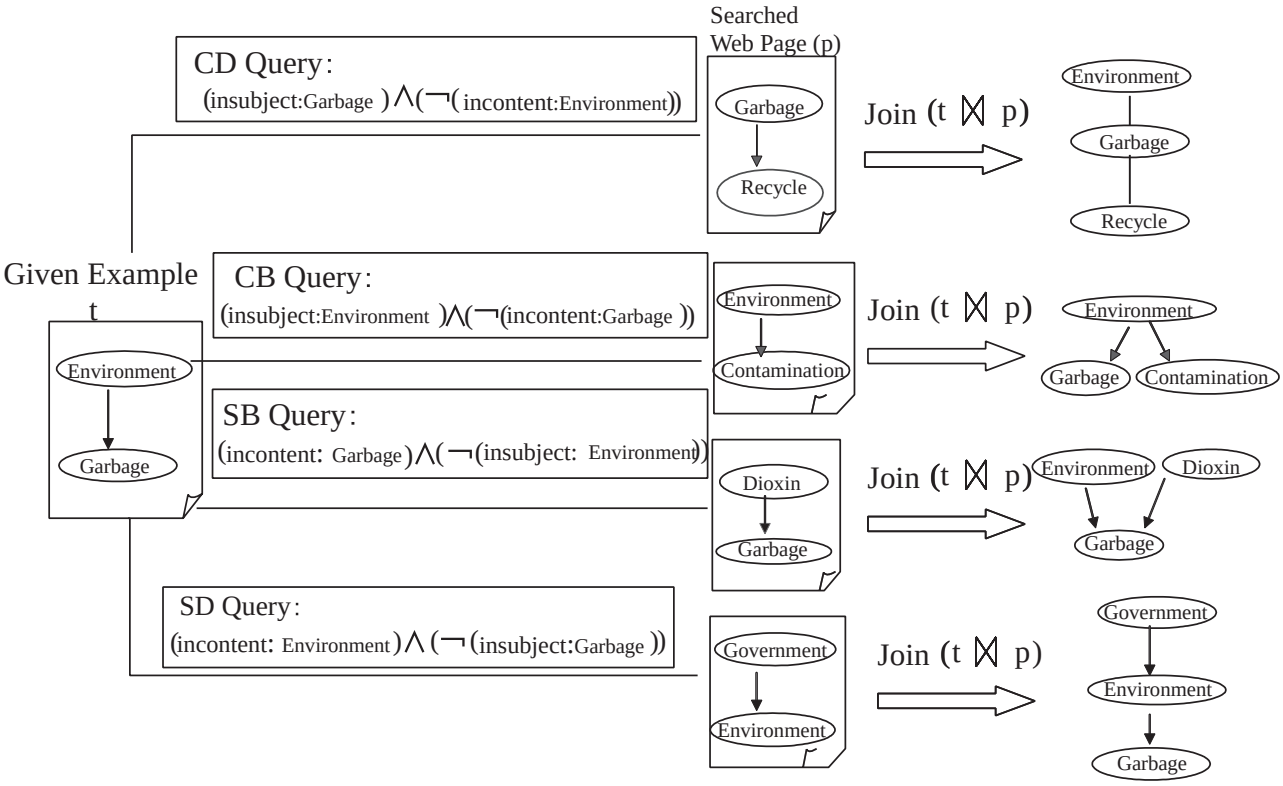

Figure 4.6: Example of CD, SD, CB and SB Queries

SB and CB queries are based on the join such that two topic structures have same subject-terms or content-terms. Such join can broaden the coverage of information.

Hereafter, let the topic structure t of a given example is $(\{s_1, s_2, ..., s_n\}, \{c_1, c_2, .., c_m\}), m \geq 1, n \geq 1$, where s_i and c_i stand for a subject term and a content term, respectively. "insubject" and "incontent" mean the following terms should be the respective subject terms and content terms of a topic structure contained in the searched web page. "$\wedge$" and "$\vee$" stand for logical AND and logical OR, respectively. "$\neg$" means the logical NOT. For example, the query $(insubject : k_1 \wedge k_2) \wedge (\neg(incontent : k_3 \wedge k_4))$ means that k_1 and k_2 should be the subject terms of a topic structure contained in the searched web page, and k_3 and k_4 should not appear as its content terms.

1. Content-Deepening Query (Q_{d_c}): For the topic structure t, by content-deepening query, we find web pages whose topic structures' subject-terms appear as the content-terms of t. Moreover, in order to avoiding the joining result ϕ, we exclude the web pages whose topic structures' content terms appear as subject terms of t.

$$Q_{d_c} = (insubject : c_1 \wedge c_2 \wedge ... \wedge c_m) \wedge (\neg(incontent : s_1 \vee s_2 \vee ... \vee s_n)) \tag{4.20}$$

2. Subject-Deepening Query (Q_{d_s}): For the topic structure t, by subject-deepening query, we find web pages whose topic structures' content-terms appear as the subject-terms of t and

their subject terms do not include any subject term of t.

$$Q_{d_s} = (incontent : s_1 \wedge s_2 \wedge ... \wedge s_n) \wedge (\neg(insubject : c_1 \vee c_2 \vee ... \vee c_m)) \tag{4.21}$$

3. Subject-Broadening Query (Q_{b_s}): By subject-broadening query, we find the web page containing a topic structure which has the same content-terms with t. Moreover, the all subject terms of t should not appear as the subject terms of the topic structure included in the searched web page. Such web page gives us complementary information from different perspective.

$$Q_{b_s} = (incontent : c_1 \wedge c_2 \wedge ... \wedge c_m) \wedge (\neg(insubject : s_1 \wedge s_2 \wedge ... \wedge s_n)) \tag{4.22}$$

4. Content-Broadening Query (Q_{b_c}): The topic structure of the web page searched by content-broadening query has the same subject-terms with t. But it should not contain the all content-terms of t as its content terms. Such web page describes the same subject of the given example from different perspective.

$$Q_{b_c} = (insubject : s_1 \wedge s_2 \wedge ... \wedge s_n) \wedge (\neg(incontent : c_1 \wedge c_2 \wedge ... \wedge c_n)) \tag{4.23}$$

Figure 4.6 shows some examples of these queries and the joining between the searched page and the given topic structure.

To distinguish from similar information retrieval, we have not defined query for searching similar information of a given example. Moreover, we exclude the similar web pages which include same topic structure as the given one by using the negative condition part. Here, negative condition part is the sub-query starting from "¬" in the last half of each query. On the other hand, the front half is called positive condition part. By the negative condition part, we may miss some valuable information but we can reduce the search results to get target one quickly. We will discuss this issue in our evaluation later.

In our current work, we use the "∧" to form the positive condition part. Actually, it is too strict to acquire enough candidate web pages at times. In such cases, we could use "∨" instead of "∧" to form the queries.

4.5.2 Ranking by Complementarity Degree

For a given topic structure, we can form four kinds of queries described above. For each query, we may acquire more than one web page. To select the most complementary one, we use the notion of complementarity degree to rank the searched web pages.

Let the reduced topic structure set of a searched web page p be $\{t_1, t_2, ..., t_n\}$. Let the topic structure of the given example is t. The complementarity degree of p can be computed as follows:

$$com(p) = \sum_{i=1}^{i=n} comple(t, t_i) \tag{4.24}$$

where, $com(p)$ stands for complementarity degree of page p to the given example.

The web page p which has the highest complementarity degree ($com(p)$), will be selected as the complementary web page of the given example (web page, video, etc.). In addition, we also consider well the ranks returned by Google. For instance, if two web pages have the same complementarity degree, we will select the one which has higher rank returned by Google as the complementary web page.

4.6 Evaluation

In this section, we describe four evaluations for our complementary information retrieval mechanism. In these evaluations, we implemented the queries by using GoogleAPI[Google Web APIs, 2003]. In evaluation *I*, we evaluated the topic detection and topic structure extraction methods. In evaluation *II*, we evaluated the complementary information retrieval method. In evaluation *III*, we compared the complementary information retrieval and similar information retrieval methods. We also discuss the difference of the four kinds of queries defined in 4.5.1. Finally, we discuss the effects of the negative condition part on discovering complementary information.

We used closed captions (in Japanese) of NHK News 7 (a well-known news TV-program in Japan) during nine month period (from September, 2002 to May, 2003) to build the co-occurrence relationship dictionary. The average size of closed captions is 26.53 KB per day. At first, we manually segmented the closed captions during one month period (September, 2002) and generated a preliminary topic collection to build a preliminary co-occurrence relationship dictionary. Then, we used such preliminary dictionary to process the rest closed captions to update the topic collection and co-occurrence relationship dictionary. We used ChaSen[ChaSen, 2003] for Japanese morphology analysis and only the nouns as keywords for further processing. To exclude stop words, we built a stop word dictionary which contains 593 terms in English and 347 terms in Japanese. Based on some preliminary evaluation results, we set the thresholds θ and Θ to be 0.15 and 0.28. θ and Θ are used to topic detection as we described before.

4.6.1 Evaluation *I*: Evaluation of Topic Extraction from Closed Captions

We used closed captions of NHK News 7 programs during 85 days (from May, 2003 to July 2003) as the evaluation data. Totally, we detected 3068 topics from these closed captions. We extracted 2 subject-terms and 3 content-terms for each topic as its topic structure.

If the rate of the words which describe one event among all words of the segmented closed captions is greater than 0.8, we say that topic is well-detected. Also, if the topic structure extracted by our method includes more than 4 terms that suit to represent the topic, we say that topic structure is well-extracted. Based on these criterions, 2460 topics and 2129 topic structures were detected and extracted well, respectively. The precision ratios (the ratio of well-detected topics (well-extracted topic structures) among total topics (topics structures)) were 0.802 and 0.694, respectively. Moreover, 1893 topics were well-segmented and their topic structures were well-extracted. We manually analyzed these closed captions and found that 3506 topics should be detected from them. For this, the recall ratio of topic detection was 0.702. Recall ratio is the rate of successfully detected topics among topics which should be detected.

There were 608 failures in topic detection. By their failure patterns, we classified them as

follows.

- 349 cases failed in detecting topics describing new events. Here, new event means it had few similar ones in the pre-build topic collections. We could not compute well the co-occurrence relationships between terms contained in the closed captions which describe such event.
- 201 detected topics described more than two events. The parameters used to topic detection and co-occurrence relationship dictionary are two of the considerable reasons.
- The closed captions of detected topics were too short. They contained less than two sentences or the time interval between the fist and last time receiving data was less than 10 seconds. We think they were not enough to describe one event. Actually, in NHK News 7, only the speaking words of news caster are available on the closed captions service. Such limitation caused this kind of failures. There were 43 such failures.
- 15 failures were caused by the typos of closed captions.

On the other hand, there were 939 failures in topic structure extraction. We classified them by their considerable causes.

- There were 503 failures caused by failure of topic detection.
- In the segmented closed captions which were considered to describe one topic, there may be some noise sentences which describe other event or activity unrelated to the main event. We might extract the keywords from such noise sentences to be subject or content terms. There were 158 such failures.
- The word which was more suitable for subject-term had small term-frequency and we could not extract it to be keyword contained in topic structure. There were 113 such failures. For example, proper nouns such as region name, organization name and a person's name often appear once, but they are suitable to be subject or content terms sometimes.
- There were 87 failures caused by the division of a word such as a person's name, organization name. For example, "Redsocks" (in Japanese) might be divided into "red" (in Japanese) and "socks" (in Japanese) by ChaSen and they may be extracted to be subject or content terms at the same time.
- Because we distinguished a word from its synonym, sometimes we extracted them currently to be subject or content-terms. It is not appropriate and there were 78 such failures.

Actually, our topic detection and topic structure extraction methods are not enough for closed captions which describe new event because they depend on co-occurrence relationships between words. In our evaluation, there were 28 days closed captions which had been used to build the co-occurrence relationship dictionary. For these data, the precision ratio (both topic detection and topic structure extraction succeeded) was 0.735. On the other hand, the precision ratio of

Table 4.1: Query Used in Evaluation *II*

	5-keys query	3-keys query
Q_{b_s}	$intext:c_1\ intext:c_2\ intext:c_3\ -allintitle:s_1\ s_2$	$intext:c_1\ intext:c_2\ -intitle:s_1$
Q_{d_s}	$intext:s_1\ intext:s_2\ -intitle:c_1\ -intitle:c_2\ -intitle:c_3$	$intitle:s_1\ -intitle:c_1\ -intitle:c_2$
Q_{b_c}	$intitle:s_1\ intitle:s_2\ -allintext:c_1\ c_2\ c_3$	$intitle:s_1\ -allintext:c_1\ c_2$
Q_{d_c}	$intitle:c_1\ intitle:c_2\ intitle:\ c_3\ -intext:s_1\ -intext:\ s_2$	$intitle:c_1\ intitle:c_2\ -intext:s_1$

Table 4.2: Evaluation Results of Complementary Information Retrieval Mechanism

	topic structure	valid-5-keys-queies	valid-3-keys-queires	relevant pages	precision ratio
Q_{b_s}	88	88	0	62	0.705
Q_{d_s}	88	81	5	55	0.625
Q_{b_c}	88	45	41	45	0.511
Q_{d_c}	88	23	43	43	0.506

the rest data was 0.526. From the results, it is obviously that co-occurrence relationship plays an important role in our methods.

The considerable approaches to improve the quality of co-occurrence relationship dictionary are described as follows.

- We can update the dictionary more frequently.
- We can use web search to discover co-occurrence relationship of words. For example, we can search the Web and use the results number as that of topics containing the word(s) used in search query to build the co-occurrence relationship dictionary.

In addition, to improve our topic detection and topic structure extraction methods, we could use the idf (inverse document frequency) value, not only the co-occurrence relationship and tf (term frequency) value. We think it is able to avoid the failures such as that candidate subject-term had small term frequency and segmented closed captions described more than two events.

4.6.2 Evaluation *II*: Evaluation of Complementary Information Retrieval

(a) Complementary Retrieval Without Ranking by Complementarity Degree

We used 3 days (May 28, 2003, June 20, 2003 and July 20, 2003) videos of NHK News 7 and their closed captions to extracted their topic structures as the given examples in this evaluation. We detected 88 topics and extracted 2 subject terms and 3 content terms for each topic.

In this evaluation, based on the query definitions described in section 4.5.1, we generated four kinds of queries for each topic structure and issued them to Google through Google API service[Google Web APIs, 2003]. Moreover, for each kind of query, at first, we generate a query (called 5-keys query) using the topic structure containing 2 subject-terms and 3 content-terms. If there is no results returned from Google, we then generate a query (called 3-keys query) using a simpler topic structure which contains 1 subject-term and 2 content-terms.

For a topic $t = (\{s_1, s_2\}, \{c_1, c_2, c_3\})$, the four kinds of queries issued to Google are shown in **Table** 4.1. Actually, we implemented these queries by using the following terms of Google API: "allintext", "alltitle", "intitle", "allintitle", "−". In Google, "allintext" restricts the results to web pages with all of the query terms following it in the text. "allintitle:" restricts the results to those with all of the query terms following it in the title. If we prepended "intitle:" to a query term, Google search restricts the results to documents containing that word in the title. On the other hand, if we prepended "intext:" to a query term, Google search restricts the results to web pages containing that word in the text.

We limited our search domain in news web sites. Because many news articles describe only one topic, this limitation also means that each searched web page has only one topic. It is not necessary to extract multiple topic structures from such web page. Thus, the complementarity degrees of these searched web pages to the given example should be same. Therefore, in this evaluation, we did not compute the complementarity degree of each searched web page. For each valid query which has more than one result returned from Google, we used the top result ranked by Google as the complementary web page.

We selected relevant complementary web pages by considering the relationship between those pages and the video (the given example): if it related to the video and provided some supplementary information, we regarded it as relevant result. Here, providing supplementary information means that it can broaden (or deepen) the subject (or content) of the video. **Table** 4.2 shows the evaluation results. Although there were some limitations, from the evaluation results, we could say that our retrieval mechanism is useful for finding the web pages which can provide supplementary information.

We noticed that if the query was based on a topic structure containing some proper nouns, the search results were better. It means that the proper nouns play an important role on topic structure based web retrieval.

Moreover, in our evaluation environment, since the corresponding closed captions had been received, the average time delay for presenting the related web page of a video was 2.85s. This shows that it is possible to develop a system which synchronizes the TV-program and its related web pages online, such as our application system WebTelop described later.

The following approaches are considerable to improve precision ratio of complementary retrieval mechanism:

- improvement of topic detection and topic structure extraction methods. It had gotten better results that a query generated from well-extracted topic structure. In particular, if a topic structure had some proper nouns, the search results were better.
- ranking the search results based on complementarity degree. We will show some evaluation results about this approach later.

(b) Filtering Based on Complementarity Degree

In evaluation *II*(b), we built a filter based on complementarity degree to select the complementary web pages of a given example from the search results. The given examples and the

Table 4.3: Evaluation Results of Filtering Based on Complementarity Degree

	recall ratio	precision ratio
SB Query	0.753	0.856
SD Query	0.639	0.753
CB Query	0.705	0.825
CD Query	0.679	0.857

Table 4.4: Comparison Evaluation Results of Similar Information Retrieval and Complementary Information Retrieval (S: average similarity between search results sets; CR: average number of common search results; Num: average number of search results.)

	CD:SIM	SD:SIM	CB:SIM	SB:SIM
S	0.097	0.109	0.184	0.149
CR	0	0.125	1.568	0.318
Num	CD: 5.11, CB:7.19, SB:9.90, SD:7.71, SIM：6.1 件			

search results of evaluation *II* (a) were used.

We computed these web pages' complementarity degrees by the method described in section 4.5.2 ($\alpha = \beta = 0.5$). Our filter selected the web pages which have higher complementarity degree than the threshold ($= 0.5$) as the complementary web pages. The evaluation results are shown in **Table** 4.3. The recall ratio is the proportion of the relevant web pages selected by using complementarity degree to all relevant web pages judged by a user. The precision ratio is the proportion of the relevant web pages selected by using complementarity degree to all select web pages. The recall ratios were not so good because that there were many web pages containing more image, video than pure text and then we would fail in computing their complementarity degrees.

Furthermore, if we select the web pages which have the highest complementarirty degree as the complementary web page of a given example, comparing to evaluation *II*(a), the precision ratios of SB, SD, CB and CD queries will be improved to be 0.807, 0.693, 0.613 and 0.688, respectively. It shows that the proposed notion, complementarity degree, is useful for picking out the more complementary web pages from the search results.

4.6.3 Evaluation *III*: Comparison Evaluations

In evaluation *III*, we compared the complementary and similar information retrieval methods. We also conducted an evaluation to discuss the difference between the four kinds of queries defined in our complementary information retrieval mechanism.

In evaluation *III*, the topic structures used to generate queries are as same as evaluation *II* (a). For each query, we used the searched pages (max 10) returned by Google as its results. To compare the search results of different queries, we used the following criteria: (1) average similarity between results sets, (2) average number of common results, and (3) average number

Table 4.5: Comparison Evaluation Results of CD, SD, SB and CB Queries (S: average similarity between search results sets; CR: average number of common search results; Num: average number of search results.)

	CD:SD	CD:CB	CD:SB	SD:CB	SD:SB	CB:SB
S	0.096	0.098	0.149	0.105	0.134	0.132
CR	0	0	0.0568	0.193	0.171	0.091
Num	CD: 5.11, CB:7.19, SB:9.90, SD:7.71					

of results.

- Average similarity between results sets: $S(A,B)$, the average similarity of results sets of query A and B, is computed based on vector space model as follows.

$$\begin{aligned} S(A,B) &= \frac{1}{N}\cdot\sum_{k=1}^{N}\frac{\sum_{i=1}^{m_k}\sum_{j=1}^{n_k} sim(p_i,p_j)}{m_k\cdot n_k} \\ sim(p_i,p_j) &= \frac{V(p_i)\cdot V(p_j)}{|V(p_i)|\times|V(p_j)|} \end{aligned} \tag{4.25}$$

  where, N is the number of queries generated from the given topic structures. In our evaluation, $N = 88$. $m_k(\leq 10)$ and $n_k(\leq 10)$ are numbers of results (searched web pages) of query a_k (an A query) and b_k (a B query) generated form topic structure k, respectively. p_i and p_j are searched web pages by query a_k and b_k, respectively. $V(p_i)$ and $V(p_j)$ are keyword vectors of p_i and p_j, respectively.

- Average number of common results: $CR(A,B)$, the average number of common results of query A and query B is computed as follows:

$$CR(A,B) = \frac{1}{N}\cdot\sum_{k=1}^{N}|r_{a_k}\cap r_{b_k}| \tag{4.26}$$

  where, r_{a_k} and r_{b_k} are results sets of a_k and b_k, respectively.

- Average number of results (searched web pages): $Num(A)$, the average number of searched web pages of query A is computed as follows.

$$Num(A) = \frac{1}{N}\cdot\sum_{k=1}^{N} m_k \tag{4.27}$$

(a) Comparison between Complementary and Similar Information Retrieval Methods

In this evaluation, we compared the similar and complementary information retrieval methods. Based on the topic structure model, we defined the similar web page to be that containing topic structure as same as the given example. We called a query used to find such similar web page

SIM query (Q_{sim}). For a given topic structure $t = (\{s_1, s_2, ..., s_n\}, \{c_1, c_2, ..., c_m\})$, the SIM query is defined as follows.

$$Q_{sim} = (insubject : s_1 \wedge s_2 \wedge ... \wedge s_n) \wedge (incontent : c_1 \wedge c_2 \wedge ... \wedge c_m) \tag{4.28}$$

The 5-keys SIM query was implemented by GoogleAPI as follows.

$$intitle : s_1 \; intitle : s_2 \; allintext : c_1 \, c_2 \, c_3 \tag{4.29}$$

The 3-keys SIM query was implemented as follows.

$$intitle : s_1 \; allintext : c_1 \; c_2 \tag{4.30}$$

We compared the search results of each complementary query (CD, CB, SB and SD query) with SIM query. The evaluation results are shown in **Table** 4.4. In addition, the number of valid (the number of searched web pages is more than one.) 5-keys and 3-keys SIM queries were 29 and 45, respectively.

The negative condition part of CD query made it to have no common search results with SIM query. Usually, the keywords appearing in the title of a web page also appear in its body. Thus, by its negative condition part, CD query will exclude such pages which may be searched by a SIM query. In addition, CD query had a few searched web pages because its positive condition part is "intitle" search query with at least 2 keywords. This feature has made the possibility smaller to return common search results of CD and SIM queries.

It is very likely that a keyword appearing in the title of a page appears in its body. However, a keyword appearing in the body does not always appear in page's title. Therefore, different from CD query, SD query had some common search results with SIM query.

By the query definitions, the web pages including the given topic structure, should be excluded from search results of CB and SB queries. However, from the evaluation results, we found that CB and SIM queries, SB and SIM queries had some common search results. This was caused by the implementation of these kinds of queries. For example, let the given topic structure be $(\{a,b\}, \{c,d,e\})$, 5-keys CB query could not exclude the web pages including the topic structure $(\{a,b\}, \{c,d\})$, which may be retrieved by 3-keys SIM query. Similarly, 5-keys SB query could not exclude the web pages including topic structure such as $(\{a\}, \{c,d,e\})$, while such web pages may be retrieved by 3-keys SIM query.

From the evaluation results, we can say that the CB, CD, SB and SD queries are different from SIM query. In other words, our complementary information retrieval mechanism differs from similar information retrieval.

(b) Comparison of CB, CD, SB and SD Queries

We also compared the search results of CB, CD, SB and SD queries. **Table** 4.5 shows the evaluation results.

From the evaluation results, it is obviously that these kinds of queries have different search results. Each of them is different form the other. However, in some cases, although the query type was different, there were a few common search results. Moreover, the difference between

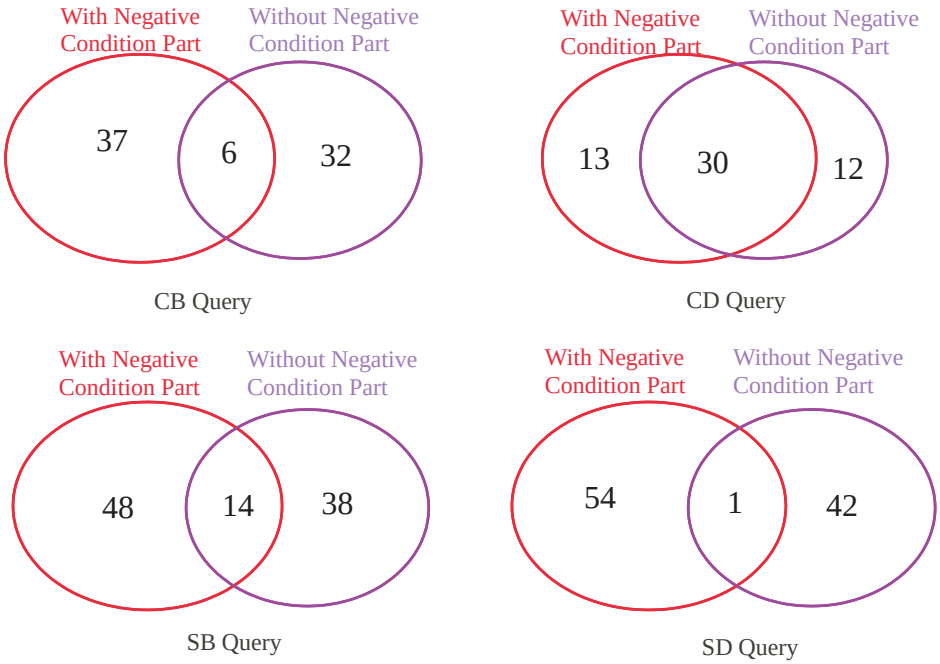

Figure 4.7: Results of Evaluation *IV*

"intitle" search and "intext" search was very big. The considerable reasons are described as follows.

- A web page may include multiple topics. Therefore, it may be an answer for different type queries.
- There are many web page not well using the *title* tag as same as we supposed. Some web pages do not use *title* tag; some web pages describe different thing such as site name in their *title* tags, not their subjects. This fashion have made it difficult to just use *title* and *body* tags to extract the web page's topic structure. Moreover, the "intitle" and "intext" searches of Google could not work well sometimes.
- The query implementation depends on GoogleAPI, and 5-keys query differs from 3-keys query.

4.6.4 Evaluation *IV*: Effects of Negative Condition Part

In evaluation *IV*, we evaluated the effects of negative condition part on complementary information retrieval. We compared the respective search results of queries with and without negative condition part. The evaluation environment was as same as experiment *II* (a). The queries not including the negation condition part are shown in **Table** 4.6. As same as evaluation *II* (a), we also used the top result returned by Google as the complementary page. We also computed the precision ratio of queries without negative condition part. If the searched complementary page is related to and can provide additional information on the given example, we say it is relevant web

Table 4.6: Query Without Negative Condition Part

	5-keys query	3-keys query
Q'_{b_s}	$intext:c_1\ intext:c_2\ intext:c_3$	$intext:c_1\ intext:c_2$
Q'_{d_s}	$intext:s_1\ intext:s_2$	$intext:s_1$
Q'_{b_c}	$intitle:s_1\ intitle:s_2$	$intitle:s_1$
Q'_{d_c}	$intitle:c_1\ intitle:c_2\ intitle:\ c_3$	$intitle:c_1\ intitle:c_2$

Table 4.7: Results of Evaluation *IV* (NCP: with negative condition part; NNCP: without negative condition part; S: average similarity between search results sets; CR: average number of common search results; Num: average number of search results.)

		S	CR	Num	Relevant Pages	Precision Ratio
CD	NCP	0.422	3.602	5.11	43	0.506
	NNCP			5.30	42	0.483
SD	NCP	0.024	0.011	7.72	55	0.625
	NNCP			10	43	0.489
CB	NCP	0.12	0.715	7.19	45	0.511
	NNCP			8.61	38	0.432
SB	NCP	0.299	3.045	9.73	62	0.705
	NNCP			9.89	52	0.591

page. The evaluation results are shown in **Table** 4.7. In addition, **Figure** 4.7 shows distribution of relevant pages searched by each kind of query. The oval stands for the relevant page set of each query. The intersection of two ovals stands for the common relevant pages for the same given topic structures. From the evaluation results, it is obviously that the negative condition part could improve the precision ratio of complementary information retrieval.

The search results of CD query with negative condition part were most similar to the one without negative condition part. CD query had a few search results and thus the negative condition part might have few effects on the search results.

On the other hand, the search results between SD queries with and without negative condition part were most different. The SD query without negative condition part is "intitle" search query using subject-terms of the given example. Because subject-terms appear often in the title (recall the notion of topic structure), a large number of web pages will be searched by such query. The negative condition part will reduce the search results effectively to find the target web page. However, as we discussed above, the *title* tag has not been well used in the current Web, we have a high risk to miss some valuable information by the negative condition part.

CB and SB queries should exclude the pages including topic structure as same as the given one by the negation condition parts to distinguish from similar information retrieval. However, as we discussed in evaluation *III*, the implementation depended on GoogleAPI[Google Web APIs, 2003] and difference between 5-keys and 3-keys queries would make some failures in excluding

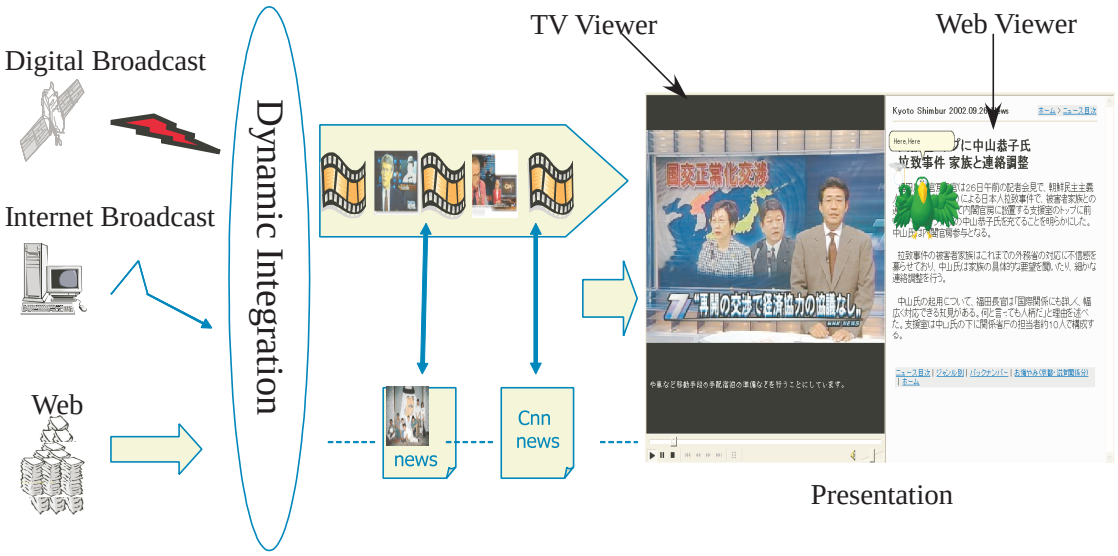

Figure 4.8: Concept of WebTelop

such pages. Therefore, in the CB and SB queries' cases, the queries with and without negative condition part had some common search results.

We assumed that subject-terms and content-terms of a web page's topic structure appear in its title and body, respectively. However, the real topic structures of web pages are more complex and there are many cases that do not satisfy our assumption. For example, a keyword appearing in the title or the body may not always be subject-term or content-term of a web page. Thus, we may miss some valuable web pages which may provide additional information on the given example. One of the approaches for avoiding such missing is that, we could search the Web by using all considerable queries and then rank the search results based on the complementarity degree.

4.7 Application System: WebTelop

4.7.1 Concept of WebTelop

The tremendous progress and spread of the information technologies have involved great changes in our lives. One of them is the digital broadcasting. Digital television combines broadcasting and computer technologies into a powerful new medium and changes the way consumers watch TV[Digital Video Broadcasting Project, 2003]. On the other hand, with the spread of broadband services that provides high speed connection to the Internet, rich content (video, music, etc.) is available to view in real time.

We propose a system for the integration and presentation of TV-program content and webpage content in real-time. In our current work, the primary source of information is the TV-program content. Web pages, which are retrieved in real time by using the complementary information retrieval mechanism, are a secondary information source. These web pages appear

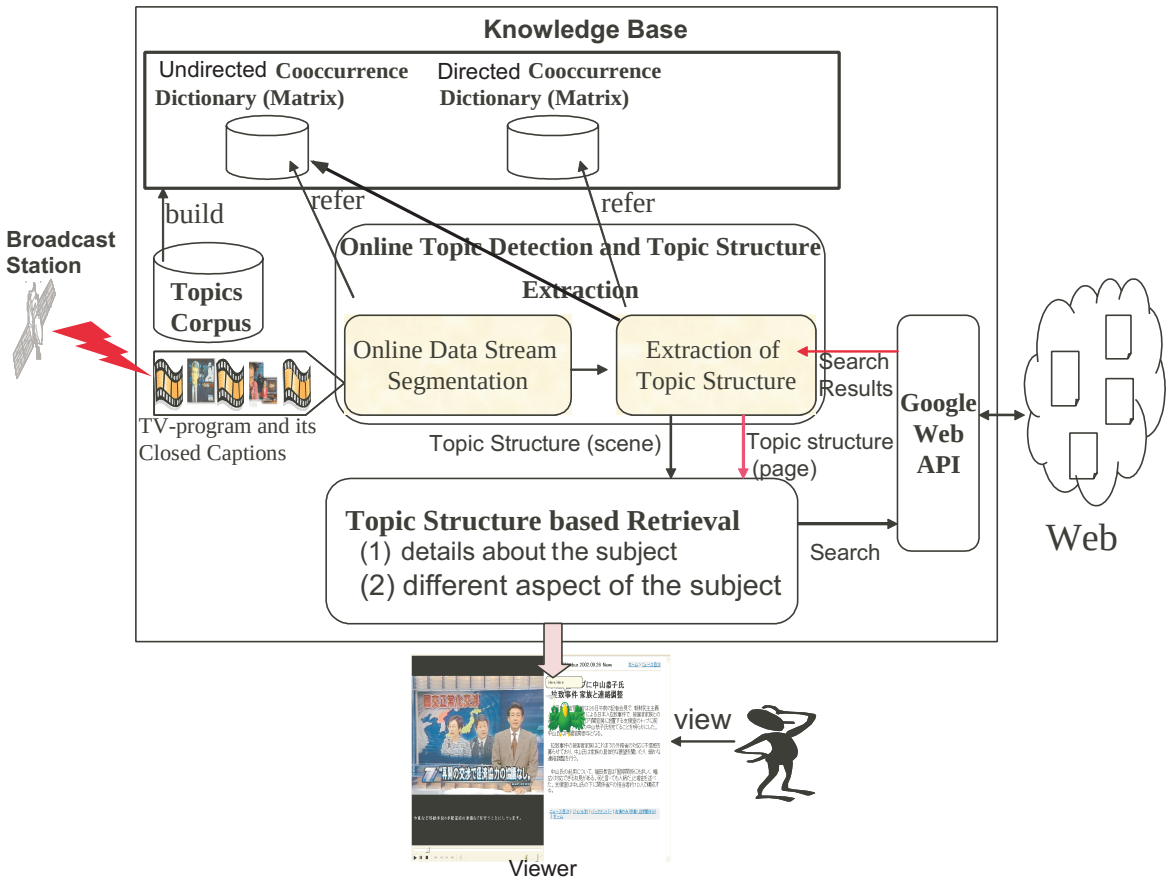

Figure 4.9: Architecture of WebTelop

concurrently with the TV program. We call the system *WebTelop* because it enables showing web pages as captions in TV programs. The web pages are not just ”similar” to the TV-program but also provide additional information on it or describe it from different perspectives. **Figure** 4.8 shows the concept of WebTelop.

4.7.2 Prototype System

In digital broadcasting environments, datacast is an add-on service in a traditional broadcasting model that is used to supplement TV programs with additional information. In this thesis, we assume that closed captions in TV programs are broadcast continuously via the datacast.

We developed a prototype system of WebTelop. **Figure** 4.9 illustrates its architecture. First, WebTelop analyzes the closed captions of a TV program to extract its topics in real time. It segments the TV program into topics[‡] by using closed caption. Then WebTelop retrieves the Internet to find complementary web page for each scene.

Although web pages provide supplemental information, we can look at them as part of augmented TV-program content. From this viewpoint, WebTelop presents TV-program content and related web pages simultaneously. In addition, a virtual character is used to help the user navigate

[‡]Hereafter, we will call these topics ”scenes”.

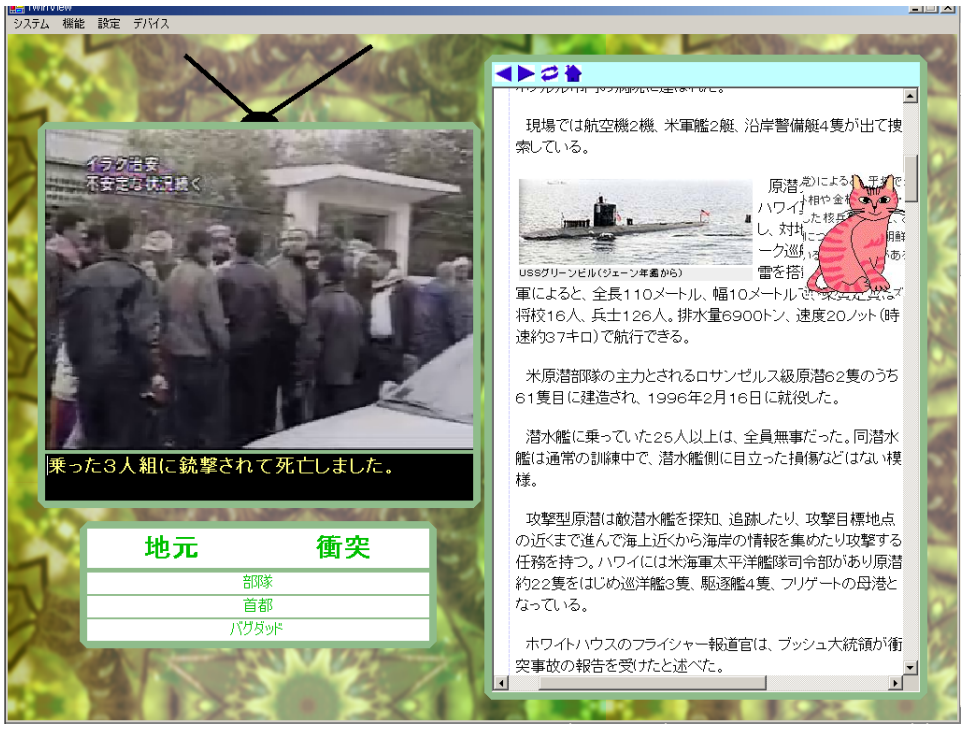

Figure 4.10: Snapshot of Concurrent Viewing with Virtual Character

through the related web pages of the TV program.

Figure 4.10 shows snapshot of the concurrent viewing of WebTelop with the virtual character. On the snapshot, the TV program is shown in the left part of the screen, and the right part shows complementary web pages. According to the topic change of TV program, displaying web pages will change automatically. The size of both parts can be adjusted automatically or manually. For example, we can increase the size of the part showing web pages when the user does not want to watch certain parts of a TV program, such as the inning changing time in a baseball game.

The virtual character was developed based on MSAgent[MSAgent, 2002]. The virtual character will guide users through the WebTelop world and will point out interesting and useful resources on the web pages by jumping to the corresponding places. Interesting parts (paragraphs) of complementary web pages are detected based on the following factors.

- the frequency of occurrence of search keywords within the paragraphs. The higher the frequency is, the more important the paragraph is.
- image files. We assume that images are more important than pure text and choose paragraphs containing images as important ones. Of course, Advertisement images are excluded.
- the similarity between each paragraph and the user profile. The paragraphs with high similarities to the user profile are considered important.

We also developed a bookmark function, by which a user can mark interested web pages

Figure 4.11: Snapshot of Concurrent Viewing with Bookmark Function

online and browse these pages later. **Figure** 4.11 is a running example of the concurrent viewing method with bookmark function.

4.8 Conclusion

For information augmentation, we proposed a novel information retrieval mechanism based on the join of topic structure. By using this mechanism, we can find the complementary information of a given web page or video. At first, we extract the topic structure of a given example (web page, video, etc.). Then, we generate some queries based on the topic-structure based join and get some candidate web pages by using Google API service. Finally, we rank these candidate web pages based on the notion of complementarity degree to select the complementary web page.

We showed some evaluation results of our proposed complementary information retrieval mechanism. From the evaluation results, we can say that the proposed mechanism is useful for searching complementary web pages which are not just similar to but also provide additional information on the given example.

Moreover, we propsed an application system WebTelop, which integrates TV-program and its related web pages in real time to augment the content of the TV-program. By using such system, a user can enjoy TV-program and its related web page concurrently to acquire information in detail or from diverse perspectives.

CHAPTER 5

CONCLUSION

With the spreading and progress of web database and digital broadcasting technology, more and more users acquire their interesting information from diverse media and diverse perspectives. To find user interesting information from the vast mount of information, information retrieval technology is effective and necessary. However, conventional information retrieval systems provide user interesting information based on matching or best matching between query (or user profile) and information sources. Moreover, they require users to form their queries clearly by using keywords. Sometimes, it is not easy to form such query and it could not satisfy the diverse information need, such as to obtain fresh, popular information, local information describing about our daily life, and additional information about our interesting topics, and so on.

As one solution of these problems, this research is focused on query-free information retrieval to select valuable information from the perspectives of time, space, and content complementation without necessary to form queries by using keywords. In this thesis, we have studied three research topics; namely, information retrieval based on temporal criteria, information retrieval based on spatial criterion, and complementary information retrieval for information augmentation.

1. Query-free information retrieval based on temporal criteria

   We proposed a novel information retrieval (filtering) method based on temporal criteria (**freshness**), **popularity** and **urgency**) which consider the worth of an article compared with previous articles. By using this method, we can discover valuable information, such as fresh information, popular information, and urgent information, from the time-series data (or data stream). We have proposed some application systems based on these temporal criteria, such as WebSCAN and the virtual TV channel system. We also estimated the effects of these temporal criteria on the information retrieval and filtering.

2. Query-free information retrieval based on spatial criterion

   We proposed a new spatial criterion **localness** to discover local information which describing about our regional and daily life. We also proposed an application system to filter

the searched web pages based on localness in order to acquire or exclude local information from the search results. We have shown some evaluation results. The evaluation results show that the spatial criterion is useful to discover information about our region and daily life.

3. Query-free complementary information retrieval for information augmentation

   We proposed a novel information retrieval method for information augmentation. It is useful to find complementary information of a given web page or video. The retrieved information is not just similar to the given web page or video, but also can provide some additional information to detail the given one or describe it from different perspective. We also showed some evaluation results on the complementary information retrieval method. In addition, we proposed an application system *WebTelop*, which integrates TV-program and its related web pages in real time to augment the content of the TV-program.

In contrast to conventional information retrieval and filtering technologies, the proposed methods go beyond similarity information retrieval. In other words, we proposed some post-similarity information retrieval methods. The information retrieved based on the proposed concepts and methods is not only similar to or exactly matching up the query or example, but also is fresh (or popular, or urgent), local and complementary information.

The proposed information retrieval methods do not require a use to form queries by using keywords. We could use the spatiotemporal criteria to discover our interesting information without any additional keyword. The complementary information is searched based on the content analysis automatically. Thus, users have no need to form a query by using keywords, too.

As the evalutaion results and the proposed application systems shown, the proposed query-free information retrieval mechanism have made it easier to acquire interesting information from multiple information sources, especially, for amateurs such as children, seniors, and so on.

We proposed some novel semantic criteria for information retrieval. These criteria are meaningful and have been defined by comparisons between the new (or given) web page or article and the others from perspectives of time, space and content complementation. Actually, it is important to consider the relationship between articles or web pages from the perspectives of time, location, subject/content, and so on. For example, two articles describe the summer festivals which were held in Kyoto and Osaka, respectively. These two articles are similar from the aspect that they are describing summer festival. However, because that the locations are different, we could say they are not similar from the aspect of location. In this thesis, we proposed some new criteria and methods for information retrieval from the perspectives of time, location and subject/content. We could use these criteria and methods in an integrated form. For example, when we compute the freshness of an article, we also can compute the difference between it and the previous articles from the perspectives of location, subject and content.

Moreover, as we mentioned in this thesis before, the criteria, such as freshness, popularity and ubiquitousness, are relative concepts. They are depended on both the time interval and region. In other words, although an article is fresh (or popular or local) in area A and time scope τ, it may be not fresh (or popular or local) in area A' and time scope τ'. Additionally, freshness and popularity

are time-varying concepts because that news articles have their valid times. For example, an article which described hot information in last year is not popular now. It is very important and necessary to select and collect well the comparison articles to compute these criteria. We will discuss these issues in our future work.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor, Professor Katsumi Tanaka at Kyoto University, for his continuous guidance, valuable advice, and helpful discussions. He also watched my abroad life in Japan warmly.

I gratefully acknowledge valuable comments of other members of my thesis committee, Professor Yahiko Kambayashi and Professor Susumu Yoshida at Kyoto University. I am deeply indebted to my adviser, Professor Syojiro Nishio at the Osaka University for his kind advice and useful discussions.

I owe my warm gratitude to Association Professor Kazutoshi Sumiya at Kyoto University for giving me advice in carrying out this research and Miss Kaori Sato, Miss Mari Suzuki, Miss Maiko Fujita, Miss Mayumi Nakata, and Miss Emiko Matsumoto for support as the staff of the laboratory.

I also owe my warm gratitude to Dr. Akiyo Nadamoto (CRL), Mr. Shinsuke Nakajima and Mr. Koji Zettsu (Graduate School of Kyoto University) for discussion, encouragements, and friendship.

I owe my kindest thanks to Mr. Shinya Miyazaki (NS Solution), Miss Chiyako Matsumoto (Canon), Mr. Takayuki Yumoto (Graduate School of Kyoto University), Mr. Hiroyuki Kondo (NHK), members of Professor Tanaka's laboratory, and all the people who gave me useful comments on the research in workshops, symposiums, and conferences for discussions, cooperation, and friendship.

Lastly, I am most grateful to my wife WU Ying for her love, support and encouragements. I also would like to thank my father MA Wanzhang, my mother XIE Guizhen and my brother MA Feng for their helping and encouragements.

BIBLIOGRAPHY

[Acharya *et al.*, 1997] Swarup Acharya, Michael Franklin, and Stanley Zdonik. Balancing push and pull for data broadcast. In *proceedings of ACM SIGMOD '97*, pages 183–194, 1997.

[Aksoy *et al.*, 1998] Demet Aksoy, Mehmet Altinel, Rahul Bose, Ugur Cetintemel, Michael Franklin, and Stan Zdonik. Research in data broadcast and dissemination. In *proceedings of 1st International Conference on Advanced Multimedia Content Processing (AMCP'98)*, pages 196–210, 1998.

[Babcock *et al.*, 2002] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. *Proceedings of the 2002 ACM Symp. on Principles of Database Systems (PODS 2002)*, pages 1–16, 2002.

[BAC, 1995] Backweb tecnologies, 1995. http://www.backweb.com.

[Brewingtion and Cybenko, 2000] Brian E. Brewingtion and George Cybenko. How dynamic is the web? In *proceedings of WWW9*, pages 264–292, 2000.

[Buyukkokten *et al.*, 1999] Orkut Buyukkokten, Junghoo Cho, Hector Garcia-Molina, Luis Gravano, and Narayanan Shivakumar. Exploiting geographical location information of web pages. In *WebDB (Informal Proceedings)*, pages 91–96, 1999.

[C3Project, 2001] C3Project. http://www-db.stanford.edu/c3/c3.html, 2001.

[Carney *et al.*, 2002] Donald Carney, Ugur Cetintemel, Mitch Cherniack, Christian Convey, Sangdon Lee, Greg Seidman, Michael Stonebraker, Nesime Tatbul, and Stanley B. Zdonik. Monitoring streams - a new class of data management applications. *Proceedings of VLDB02*, pages 215–226, 2002.

[ChaSen, 2003] ChaSen. http://chasen.aist-nara.ac.jp/index.html.en, 2003.

[Chawathe and Garcia-Molina,] Sudarshan S. Chawathe and Hector Garcia-Molina. Meaningful change detection in structured data. In *proceedings of SIGMOD'97*, page 26.

[Consortium, 1998] World Wide Web Consortium. Extensible markup language(xml) 1.0, 1998. http://www.w3.org/TR/REC-xml.

[Cyveillance, 2000] Cyveillance. http://www.cyveillance.com/, 2000.

[Digital Video Broadcasting Project, 2003] Digital Video Broadcasting Project. http://www.dvb.org, 2003.

[Douglis *et al.*, 1996] Fred Douglis, Thomas Ball, Yih-Farn Chen, and Eleftherios Koutsofios. Webguide: Querying and navigating changes in web repositories. In *Proceedings of WWW5*, pages 1335–1344, 1996.

[Egnor, 2002] Daniel Egnor. Google programing contest, 2002.

[Google Web APIs, 2003] Google Web APIs. http://www.google.com/apis/, 2003.

[Google, 2003] Google. http://www.google.com, 2003.

[Guha *et al.*, 2002] Sudipto Guha, H.V.Jagadish, and Nick Koudas. Aproximate xml joins. *Proceedings of ACM SIGMOD 2002*, pages 287–298, 2002.

[Guttman, 1984] Antonin Guttman. R-trees: A dynamic index structure for spatial searching. *Proc. ACM SIGMOD Conference on Management of Data*, 14(2):47–57, 1984.

[Hayashi *et al.*, 1999] M. Hayashi, H. Ueda, and T Kurihara. Tvml (tv program making language) - automatic tv program generation from text-based script-. In *Proc. of Imagina'99*, 1999.

[Henzinger *et al.*, 2003] Monika Henzinger, Bay-Wei Chang, Brian Milch, and Sergey Brin. Query-free news search. *Proceedings of The Twelfth International World Wide Web Conference*, 2003.

[Inc., 2003] Miniwatts International Inc. http://www.internetworldstats.com, 2003.

[Ishida, 2002] Toru Ishida. Digital city kyoto: Social information infrastructure for everyday life. In *Communications of the ACM, Vol.45, No.7*, pages 76–81, 2002.

[James Allan, 1998] Victor Lavrenko James Allan, Ron Papka. On-line new event dectection and tracking. In *Proceedings of SIGIR'98*, pages 37–45, 1998.

[Kamba *et al.*, 1997] T. Kamba, H.Sakagami, and Y.Koseki. Automatic personalization on push news service. In *W3C Push Workshop*, 1997. http://www.w3.org/architecture/ 9709_Workshop/ paper02/paper02.html.

[Lee *et al.*, 2002] Ryong Lee, Hiroki Takakura, and Yahiko Kambayashi. Virtual query processing for gis with web contents. In *proceedings of the 6th IFIP working conference on visual database systems*, 2002.

[Liu *et al.*, 2000] Ling Liu, Calton Pu, and Wei Tang. WebCQ-detecting and delivering information change on the web. In *proceedings of CIKM'00*, 2000.

[MACHIgoo, 2003] MACHIgoo. http://machi.goo.ne.jp, 2003.

[Maeda *et al.*, 1997] Harumi Maeda, Kazuto Koujitani, and Toyoaki Nishida. Information reorganization using associative structures. In *IPSJ Journal, Vol. 38, No. 3*, pages 616–625, 1997.

[Mani *et al.*, 1997] Inderjeet Mani, David House, Mark Maybury, and Morgan Green. Towards content-based browsing of broadcast news video. In *Intelligent multimedia information retrieval, Mark Maybury, ed., chapter 12*, pages 241–258, 1997.

[Marimba, 1998] Marimba. Castanet, 1998. http://www.marimba.com.

[Matsukura *et al.*, 2001] Takeshi Matsukura, Hiroyuki Kondo, Yoichi Hirata, and Katsumi Tanaka. Discovery of semantic relationships among web pages based on web topic structures. *Proceedings of 9th IFIP 2.6 Working Conference on Database Semantics*, pages 184–199, 2001.

[Maybury, 1997] Mark Maybury. *Intelligent Multimedia Information Retrieval*. AAAI Press and MIT Press, 1997.

[Microsoft, 1999] Microsoft. http://www.microsoft.com/xml, 1999.

[Ministry of Public Management, Home Affairs Posts and Telecommunications, Japan, 2003] Ministry of Public Management, Home Affairs Posts and Telecommunications, Japan. Information and communications in Japan, 2003.

[MIS2, 2002] MIS2. http://www.kokono.net/, 2002.

[Mishra and Eich, 1992] Priti Mishra and Margaret H. Eich. Join processing in relational databases. *ACM Computing Surveys Vol.24, No.1*, pages 63–112, 1992.

[Miura *et al.*, 1998] Nobuyuki Miura, Katsumi Takahashi, Seiji Yokoji, and Kenichi Shima. Location oriented information integration ～ mobile info search 2 experiment ～. *The 57th National Convention of IPSJ*, 3:637–638, 10 1998.

[MSAgent, 2002] MSAgent. http://www.microsfot.com/msagent, 2002.

[Munakata *et al.*, 2000] Koichi Munakata, Masatoshi Yoshikawa, and Shunsuke Uemura. Acquiring data combinations from periodically generated data sequences based on freshness and synchronousness. In *IPSJ Transactions on Databases (TOD5)*, pages 140–153, 2000.

[Murakami and Hirata, 2001] Harumi Murakami and Takashi Hirata. Information acquistion and organization from www (in japanese). In *IPSJ Technical Reports FI-142-23*, pages 167–174, 2001.

[.net, 2002] Microsoft .net. http://www.microsoft.com/net/, 2002.

[NetMind, 2001] NetMind. http://www.netmind.com, 2001.

[Oyama and Tanaka, 2003] Satoshi Oyama and Katsumi Tanaka. Exploiting document structures for comparing and exploring topics on the web. *Proceeding of the 12th Internaional World Wide Web Conference (WWW2003) (poster tracks)*, 2003.

[PointCast Network, 1999] PointCast Network. http://www.pointcast.co.jp, 1999.

[Ramakrishnan and Dayal, 1998] Satish Ramakrishnan and Vibha Dayal. The PointCast network. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 27(2):520, 1998.

[SALTON, 1968] G. SALTON. Automatic information organization and retrieval, 1968.

[Shapiro, 1998] David M. Shapiro. Push-based web filtering using pics profiles, April 1998. http://www.w3.org/TandS/Public/Theses/ DavidShapiro/thesis-dshapiro.html.

[Shasha *et al.*, 2002] Dennis Shasha, Jason T.L. Wang, and Rosalba Giugno. Algorithmics and applications of tree and graph searching. *Proceedings of the 2002 ACM Symp. on Principles of Database Systems (PODS 2002)*, pages 39–52, 2002.

[Sullivan and Heybey, 1998] Mark Sullivan and Andrew Heybey. Tribeca: A system for managing large databases of network traffic. *Proceedings of the 1996 USENIX Annual Technical Conference*, pages 13–24, 1998.

[Sumiya and Miyabe, 1999] Kazutoshi Sumiya and Yoshiyuki Miyabe. Models and systems for data broadcast. In *IPSJ Transactions on Databases (TOD4)*, pages 141–157, 1999.

[Takeda, 2000] T. Takeda. The latitude / longitude position database of all-prefectures cities, towns and villages in japan, 2000.

[Theater, 1997] SiteCrusie Theater. http://www.sitecruise.com, 1997.

[TopicMap.org, 2003] TopicMap.org. http://www.topicmap.org, 2003.

[Tucker *et al.*, 2002] Pete Tucker, David Maier, Tim Sheard, and Leonidas Fegaras. Punctuating continuous streams. *Technical Report, Oregon Graduate Institute*, 2002.

[TVML, 2003] TVML. http://www.strl.nhk.or.jp/tvml/index.html, 2003.

[Wactlar, 2000] Howard D. Wactlar. Informedia - search and summarization in the video medium. *Proceedings of Imagina 2000 Conference*, 2000.

[Wayne, 2000] Charles L. Wayne. Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation. *Proceedings of the Language Resources and Evaluation Conference (LREC) 2000*, pages 1487–1494, 2000.

[XSL, 2001] XSL. http://www.w3.org/tr/xsl, 2001.

[Yahoo!, 2003] Yahoo! http://www.yahoo.co.jp, 2003.

[Yahoo!regional, 2003] Yahoo!regional. http://local.yahoo.co.jp, 2003.

[Yan and Garcia-Molina, 1995] Tak W. Yan and Hector Garcia-Molina. SIFT — a tool for wide-area information dissemination. In USENIX Association, editor, *Proceedings of the 1995 USENIX Technical Conference: January 16–20, 1995, New Orleans, Louisiana, USA*, pages 177–186, Berkeley, CA, USA, January 1995. USENIX.

[Yiming Yang, 1998] Jaime Carbonell Yiming Yang, Tom Pierce. A study on retrospective and on-line event detection. In *Proceedings of SIGIR'98*, pages 28–36, 1998.

[Zaniolo *et al.*, 1997] Carlo Zaniolo, Stefano Ceri, Christos Faloutsos, Richard T. Snodgrass, V. S. Subrahmanian, and Roberto Zicari. *Advanced Database Systems*. The Morgan Kaufmann, 1997.

[Zloof, 1977] M. Zloof. Query-by-example: A data base language. *IBM Systems Journal, 16, 4*, pages 324–343, 1977.

PUBLICATIONS

Journal and International Conference Papers

- Qiang Ma, Katsumi Tanaka: Topic-Structure Based Complementary Information Retrieval for Information Augmentation. Lecture Notes in Computer Science (APWeb04), LNCS 3007, pp. 594-605, 2004.
- Qiang Ma, Chiyako Matsumoto, Katsumi Tanaka: A Localness-Filter for Searched Web Pages, Lecture Notes in Computer Science(APWeb03), LNCS 2642, pp. 525-536, 2003.
- Qiang Ma, Katsumi Tanaka: WebTelop: Dynamic TV-content Augmentation by Using Web Pages, Proceedings of IEEE International Conference on Multimedia and Expo (ICME2003), Vol.2, pp.173-176, 2003
- Qiang Ma, Shinya Miyazaki, Katsumi Tanaka: WebSCAN: Discovering and Notifying Important Changes of Web Sites, Lecture Notes in Computer Science (DEXA01), LNCS 2113, pp.587-598, 2001
- Qiang Ma, Kazutoshi Sumiya, Katsumi Tanaka: Information Filtering Based on Time-series Features for Data Dissemination Systems(in Japanese). IPSJ Transactions on Databases (TOD7), pp.46-57, 2000
- Qiang Ma, Hiroyuki Kondo, Kazutoshi Sumiya, Katsumi Tanaka: Virtual TV Channel: Filtering, Merging, and Presenting For Internet Broadcasting Channels, Proceedings of WOWS'99, pp.32-43,1999
- Katsumi Tanaka, Kazutoshi Sumiya, Akiyo Nadamoto, Qiang Ma: Broadcasting and Databases, Nontraditional Database Systems, the Information Processing Society of Japan and Taylor and Francis Books Ltd., ISBN 0-415-30206-4, pp47-62, 2002.
- Chiyako Matsumoto, Qiang Ma, Katsumi Tanaka: Web Information Retrieval Based on the Localness Degree, Lecture Note of Computer Science(DEXA02), LNCS 2453, pp.172-181, 2002
- Shinya Miyazaki, Qiang Ma, Katsumi Tanaka: WebSCAN: Content-based change discovery and broadcast-notification of Web sites (in Japanese), IPSJ Transactions on Databases (TOD10), pp.96-107, 2001

- Akiyo Nadamoto, Qiang Ma, and Katsumi Tanaka: Concurrent Browsing of Bilingual Web Sites By Context-Synchronization and Difference, Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE2003), pp. 189-199, 2003

- Takayuki Yumoto, Qiang Ma, Kazutoshi Sumiya, and Katsumi Tanaka: A Dynamic Content Integration Language for Video Data and Web Contents, Proceedings of the 4th International Conference on Web Information Systems Engineering (WISE2003), pp. 83-92, 2003

- Takayuki Yumoto, Naoki Hukino, Qiang Ma, Kazutoshi Sumiya and Katsumi Tanaka: Video-Augmented Web:Video-Augmented Web : Dynamic Integration of Video Stream into Web Pages (in Japanese), DBSJ Letters, Vol.2,No.2, pp.41-44, 2003

Symposium and Workshop Papers

Reviewed Paper

- Qiang Ma, Kazutoshi Sumiya, and Katsumi Tanaka: WebTelop: A Dynamic Integration and Presentation System of Web and Broadcasting Information. Proceedings of the seventh conference on achievements in scientific research of chineses scholars in Japan, pp. 433-440, 2002.

- Qiang Ma, Chiyako Matsumoto,and Katsumi Tanaka: Localness Degree of Web Pages and Its Applications from Page Content and Location Information, Proceedings of the seventh conference on achievements in scientific research of chineses scholars in Japan, pp. 425-432, 2002

- Qiang Ma, Kazutoshi Sumiya, and Katsumi Tanaka: Virtual Channel in Broadcast-based Information Delivery System and it's Relization with XML (in Japanese), Proceedings of DEWS 99, pp.714-722, 1999

- Naoki Fukino, Qiang Ma, Kazutoshi Sumiya and Katsumi Tanaka: Generating Football Video Summary UsingNews Article (in Japanese), Proceedings of DEWS2003, 2003

- Takayuki Yumoto, Qiang Ma, Kazutoshi Sumiya and Katsumi Tanaka: Multimedia Contents Integration Reflecting Authors' Intention (in Japanese), Proceedings of DEWS2003, 2003

- Shinya Miyazaki, Qiang Ma, Kazutoshi Sumiya and Katsumi Tanaka: Change Detection and Notification of Dynamic Web Page Generation Environments Using XML Databases (in Japanese), Proceedings of DEWS2002, 2002

- Chiyako Matsumoto, Qiang Ma, and Katsumi Tanaka: Information Filtering Basedon Localness Degree of Web pages (in Japanese), Proceedings of DBWeb2001, pp.193-200, 2001

- Shinya Miyazaki, Qiang Ma, Katsumi Tanaka: WebSCAN:Content-Based Change Discovery and Broadcast-Notification for Web sites (in Japanese), Proceedings of DBWeb2000, pp.89-96, 2000

Technical Reports

- Qiang Ma, Katsumi Tanaka:Topic-structure-based Join Operation and Its Applications (in Japanese), IPSJ Technical Reports, 2003-DBS-131, pp.153-159,2003

- Qiang Ma, Jie Lin, Kazutoshi Sumiya and Katsumi Tanaka:Description and Presentation Mechanism for Local Advertisement Content Including Appeal Function (in Japanese), IPSJ Technical Reports, 2002-DBS-129, pp.121-128, 2003

- Qiang Ma, Chiyako Matsumoto, and Katsumi Tanaka: Localness Degree of Web Pages and Its Applications from Page Content and Location Information (in Japanese), Technical Reports, 2002-DBS-128-69, pp.515-522, 2002

- Qiang Ma, Kazutoshi Sumiya, and Katsumi Tanaka: WebTelop: A Dynamic Integration and Presentation System of Web and Broadcasting Information (in Japanese), IPSJ Technical Reports, 2002-DBS-128-23, pp.169-176, 2002

- Qiang Ma, Kazutoshi Sumiya, and Katsumi Tanaka: Integration and Filtering Functions for Stream Data and Their Applications (in Japanese), IEICE Technical Reports, DE2002-6, pp.29-34, 2002

- Qiang Ma, Kazumi Tanaka:Disposing and Restructing of Stored Time-Series Articles Based on Freshness and Popularity (in Japanese), IPSJ Technical Reports, 2000-DBS-122-9, pp. 65-72, 2000

- Qiang Ma, Hiroyuki Kondo, Kazutoshi Sumiya, and Katsumi Tanaka: Virtual TV Channel: Filtering, Merging, and Presenting for internet broadcasting channels,IPSJ Technical Reports, 99-DBS-119, pp.189-194, 1999

- Takayuki Yumoto, Naoki Hukino, Qiang Ma, Kazutoshi Sumiya and Katsumi Tanaka: Video-Augmented Web:Video-Augmented Web : Dynamic Integration of Video Stream into Web Pages (in Japanese), IPSJ Technical Reports, 2003-DBS-131, pp.383-390, 2003

- Chiyako Matsumoto, Qiang Ma, Katsumi Tanaka: Information Filtering by Detecting Localness Degree (in Japanese), IPSJ Technical Reports, pp.273-280, 2001

- Chiyako Matsumoto, Qiang Ma, Katsumi Tanaka: A News Filtering Mechanism by Geographical Information (in Japanese), proceedings of the 62th national convention of IPSJ, pp.505-506, 2001

- Shinya Miyazaki, Qiang Ma, Katsumi Tanaka: A Monitoring and Change Notification System for Web Sites (in Japanese), IPSJ Technical Reports, pp.527-534, 2000

- Noda Reiko, Ma Qiang, Sumiya Kazutoshi and Tanaka Katsumi: An Information Delivery System using Temporal Infomation (in Japanese), IPSJ Technical Reports, 98-DBS-116, pp.103-110, 1998